

# UC Irvine

## UC Irvine Electronic Theses and Dissertations

### Title

Resilient Spatiotemporal Truck Monitoring Framework using Inductive Signature and 3D Point Cloud-based Technologies

### Permalink

<https://escholarship.org/uc/item/6qj8q757>

### Author

Li, Yiqiao

### Publication Date

2021

### Copyright Information

This work is made available under the terms of a Creative Commons Attribution-NonCommercial-NoDerivatives License, available at <https://creativecommons.org/licenses/by-nc-nd/4.0/>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA,  
IRVINE

Resilient Spatiotemporal Truck Monitoring Framework using Inductive Signature  
and 3D Point Cloud-based Technologies

DISSERTATION

submitted in partial satisfaction of the requirements  
for the degree of

DOCTOR OF PHILOSOPHY

in Civil and Environmental Engineering

by

Yiqiao Li

Dissertation Committee:  
Professor Stephen G. Ritchie, Chair  
Professor Amelia C. Regan  
Professor Wilfred W. Recker

2021



## **DEDICATION**

To my mother and mentors  
in recognition of their support, motivation, and patience

In Loving Memory of  
my father and my grandparents

# TABLE OF CONTENTS

	Page
<b>DEDICATION</b> .....	ii
<b>LIST OF FIGURES</b> .....	vi
<b>LIST OF TABLES</b> .....	x
<b>ACKNOWLEDGEMENTS</b> .....	xi
<b>CURRICULUM VITA</b> .....	xiii
<b>ABSTRACT OF THE DISSERTATION</b> .....	xvii
<b>CHAPTER 1 : Introduction</b> .....	1
<b>1.1 Impacts of Trucks</b> .....	1
<b>1.2 Current Truck Data Sources and Concerns</b> .....	5
<b>1.3 Proposed Solution Overview</b> .....	8
<b>1.4 Dissertation Outline</b> .....	9
<b>CHAPTER 2 : Intrusive-Sensor Solutions: Major Highway Truck Monitoring using Advanced Inductive Loops</b> .....	12
<b>2.1 Individual Truck Speed Estimation from Advanced Single Inductive Loops</b> .....	12
2.1.1 Introduction.....	12
2.1.2 Literature Review.....	16
2.1.3 Data Description .....	21
2.1.4 Model Development .....	22
2.1.5 Model Results.....	35
2.1.6 Conclusion.....	40
<b>2.2 A Deep Ensemble Neural Network Approach for FHWA Axle-based Vehicle Classification using Advanced Single Inductive Loops</b> .....	42
2.2.1 Introduction.....	42
2.2.2 Literature Review.....	45
2.2.3 Data Description .....	49
2.2.4 Model Development .....	53
2.2.5 Model Results.....	58
2.2.6 Conclusion.....	66

<b>CHAPTER 3 : Non-Intrusive-Sensor Solutions: Rural Highways Truck Monitoring using Multi-array LiDAR Sensors</b> .....	68
<b>3.1 Introduction</b> .....	68
<b>3.2 Literature Review</b> .....	69
<b>3.3 Data Description and Preprocessing</b> .....	74
3.3.1 Study Site Layout.....	74
3.3.2 Data Collection Setup.....	75
3.3.3 Data Description .....	77
3.3.4 Semi-automatic Data Labeling Method.....	79
3.3.5 Data Preprocessing.....	80
<b>3.4 Vehicle Point Cloud Registration Framework</b> .....	81
3.4.1 Point Cloud Registration.....	81
3.4.2 Vehicle Point Cloud Registration Framework.....	84
<b>3.5 FHWA Axle-based Classification using Roadside LiDAR Sensor</b> .....	95
3.5.1 Feature Extraction .....	96
3.5.2 Bootstrap Aggregating Deep Neural Network for Vehicle Classification .....	98
3.5.3 Model Results.....	100
<b>3.6 Truck Body Type Classification using Roadside LiDAR Sensor</b> .....	105
3.6.1 PointNet-based Truck Classification Model.....	105
3.6.2 Model Averaging.....	109
3.6.3 Results .....	111
3.6.4 Conclusion.....	117
<b>CHAPTER 4 : A Self-Learning Framework for Truck Classification System through the Integration of Advanced Inductive Loops and Multi-array LiDAR Sensor</b> .....	119
<b>4.1 Introduction</b> .....	119
<b>4.2 Interpreting DNN-based Vehicle Classification Model Predictions</b> .....	121
4.2.1 Human Intuition.....	121
4.2.2 Interpreting Vehicle Classification Model Predictions .....	122
<b>4.3 A Self-learning Framework for Truck Monitoring System</b> .....	127
4.3.1 The Self-learning Framework.....	127
4.3.2 System Implementation.....	132
<b>4.4 Conclusion and Discussion</b> .....	139

<b>CHAPTER 5 : Conclusion, Discussions, and Future Works</b> .....	142
<b>5.1 Conclusion</b> .....	142
<b>5.2 Future Work</b> .....	144
<b>REFERENCES:</b> .....	145

## LIST OF FIGURES

	Page
Figure 1.1 Commodity Values, Tons and Ton-miles transported by Trucks (U.S. Department of Transportation et al., 2017).....	1
Figure 1.2 Average ESALs by FHWA Vehicle Classes and Infrastructure Functional Classes (Note: Since Motorcycles, passage cars, and SUV/Pickup trucks do not significantly contribute to the ESALs, they are negligible and their ESALs are approximately 0.) (Federal Highway Administration, 2013; Schmoyer & Hu, 1998) .....	3
Figure 1.3 Percentage Estimated U.S. Average Vehicle Emission Rate per Vehicle by Vehicle Type using Gasoline and Diesel (Statistics Bureau of Transportation, 2021) Note: Light-duty vehicles: passenger cars. Light-duty trucks: trucks with two axles and four tires. Heavy-duty vehicles: trucks with more than two axles or four tires.....	4
Figure 1.4 Relationship between Each Chapter .....	9
Figure 2.1 Overview of the AISE Model.....	16
Figure 2.2 Comparison between simple moving average and progressive moving average (a) SMA with a window size of 79 (b) PMA with a window size of 79.....	23
Figure 2.3 The Optimal Window Size.....	25
Figure 2.4 The Optimal Cutting Threshold .....	25
Figure 2.5 A Raw Signature and Processed Signature (a) A Raw Signature (b) A Processed Signature .....	25
Figure 2.6 Feature Extraction.....	26
Figure 2.7 The Sum of Square Error for Different K values.....	28
Figure 2.8 The Lowest Adjusted $R^2$ for Different K values.....	29



Figure 2.9 Feature Weighting on K-means Algorithm.....	30
Figure 2.10 K-means Clustering Results.....	31
Figure 2.11 Feature Extraction for the Multi-layer Perceptron Model (a) Thirty Magnitude Features (b) Thirty Magnitude Differences Features .....	34
Figure 2.12 Testing Results for the ATISE Model.....	37
Figure 2.13 Testing Results for the ISE Model.....	37
Figure 2.14 30 Second Aggregated Speed Estimation Result using ATISE Method: (a) 2018- 7-13 (Weekday), (b) 2018-7-14 (Weekend) .....	39
Figure 2.15 30 Second Aggregated Speed Estimation Result using ISE Method: (a) 2018-7- 13 (Weekday), (b) 2018-7-14 (Weekend) .....	40
Figure 2.16 Class 9 Enclosed Van and its corresponding raw signature.....	48
Figure 2.17 Data Collection site for model training, hyperparameter tuning, and transferability testing.....	50
Figure 2.18 Signatures of different truck body types .....	52
Figure 2.19 Preprocessing and Feature Extraction .....	54
Figure 2.20 Model Structure.....	55
Figure 2.21 Learning Curve.....	56
Figure 2.22 Illustration of Bagging DNN .....	58
Figure 2.23 Correct Classification Rate across all Classes .....	60
Figure 2.24 Class 3 vs Class 5 (FHWA (Federal Highway Administration), 2014).....	64
Figure 2.25 Class 3 vs Class 8 (FHWA (Federal Highway Administration), 2014).....	64
Figure 2.26 Overlapping body type across FHWA classes.....	65
Figure 2.27 Error cases for piezoelectric sensors (Bitar & Refai, 2017) .....	66

Figure 3.1 Illustration of LiDAR Orientation .....	71
Figure 3.2 Truck Point Cloud Collection from Vertically Oriented LiDAR.....	71
Figure 3.3 Layout of the Detection Site .....	75
Figure 3.4 System Setup .....	76
Figure 3.5 Illustration of the LiDAR Sensor .....	77
Figure 3.6 The Raw Point Cloud of the Detection region .....	77
Figure 3.7 Illustration of Vehicle Body Configuration used in the study (Note: SU: Single Unit Trucks, Semi: Tractors pulling Semi-Trailer, Multi: Tractors pulling Multiple Trailers.) .....	78
Figure 3.8 Semi-automatic Data Labeling Framework.....	80
Figure 3.9 Samples of Truck Frames .....	85
Figure 3.10 Samples of Redundant Frames .....	86
Figure 3.11 Samples of Frames used for Registration.....	86
Figure 3.12 Elimination of Redundant Frames .....	87
Figure 3.13 Vehicle Point Cloud Registration Framework .....	90
Figure 3.14 Examples of Pairwise Registration (Blue: target point cloud, Yellow: Source point cloud) .....	91
Figure 3.15 Illustration of Experiment Design .....	93
Figure 3.16 Truck Object from a Single Frame (Left) vs. A Reconstructed Truck Point Cloud (Right) .....	95
Figure 3.17 Feature Extraction.....	96
Figure 3.18 Illustration of Features .....	98
Figure 3.19 Learning Curve.....	99

Figure 3.20 Normalized Confusion Matrix for the Test Set .....	100
Figure 3.21 CCR Distribution across All Classes .....	102
Figure 3.22 Overlapping Body Configurations .....	103
Figure 3.23 PointNet Architecture (Qi et al., 2017) .....	106
Figure 3.24 Point Cloud Preprocessing .....	108
Figure 3.25 Learning Curves .....	109
Figure 3.26 Prediction Variance Analysis .....	111
Figure 4.1 A Vehicle Point Cloud and A Signature from the Same 40ft Container .....	122
Figure 4.2 SHAP Values for Features from Signature Data .....	124
Figure 4.3 SHAP Values for Features from Point Cloud Data .....	125
Figure 4.4 Differences among Classes 8, 9, and 10 .....	126
Figure 4.5 Differences between Classes 11 and 12 .....	126
Figure 4.6 Self-Learning Truck Classification System .....	128
Figure 4.7 Illustration of Transfer Learning .....	130
Figure 4.8 Adaptive Transfer Learning for Truck Classification .....	131
Figure 4.9 Data Description .....	133
Figure 4.10 Model Validation .....	135
Figure 4.11 The Testing Results of the Adaptive Transfer Learning on the San Onofre dataset and the Statewide dataset .....	136

## LIST OF TABLES

	Page
Table 1.1 Current Truck Data Sources.....	5
Table 2.1 Statistical Summary of Each Linear Model.....	32
Table 2.2 MLP Performance for Different Clustering Method.....	35
Table 2.3 Evaluation of Individual Vehicle Speed Estimation on Training and Test Set.....	36
Table 2.4 FHWA-CA classification scheme definitions (FHWA (Federal Highway Administration), 2014).....	43
Table 2.5 Vehicle Class Distribution from ground truth.....	51
Table 2.6 Generic Confusion Matrix.....	59
Table 2.7 Test Result Comparison.....	61
Table 2.8 Model Comparison.....	62
Table 2.9 Confusion Matrix for Test Set.....	63
Table 3.1 Summary of LiDAR-based Vehicle Classification.....	73
Table 3.2 Experiment Results .....	94
Table 3.3 Model Comparison.....	104
Table 3.4 Results from Test Dataset.....	112
Table 3.5 Confusion Matrix for single-unit truck and passenger vehicles (SMA Approach) .....	114
Table 3.6 Confusion Matrix for a truck with Trailer(s) (SMA Approach) .....	115
Table 4.1 Model Comparison.....	138
Table 4.2 Model Validation on Groundtruth-labeled dataset vs. LiDAR-labeled dataset ....	139

## ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Prof. Stephen Ritchie who gave me a warm welcome to the transportation system engineering Ph.D. program five years ago. He has always been supportive of my research and provides me with opportunities to gain experience in teaching. I feel grateful to work with him during my Ph.D. studies.

Most importantly, I would like to express my deepest appreciation to Dr. Andre Tok. Dr. Tok has always provided me insightful comments and critiques of my research which helps me think rigorously on addressing research problems and establish a grand view of my research. He not only provided me with valuable writing and presentation tips but resources to help me improve my communication skills. He is the best mentor I have ever met.

I would like to thank all my committee members, Prof. Amelia Regan, Prof. Wilfred Recker, Prof. Zhaoxia Yu, and Prof. R. Jayakrishnan for their valuable suggestions and support of my research. I would also like to thank Prof. Wenlong Jin, Prof. Michael G. McNally, Prof. Jean-Daniel Saphores, and Michael Hyland for their transportation-related classes which equipped me with the knowledge to initiate my Ph.D. studies.

I also appreciated the kindly assistance of the administrative staff from ITS and Civil Engineering: Ms. Cam Tran, Ms. April Health, and Ms. Angie Zungia.

I would like to thank all the members of Prof. Ritchie and Dr. Tok's LiDAR research team, Dr. Zhe Sun, Koti Allu, Guoliang Feng, and Dr. Yingji Xia for their data collection and processing efforts and for providing such a relaxed work environment. I would also like to

acknowledge Dr. Suman Mitra, Dr. Kyung Hyun, Dr. Kyungsoo Jeong, Dr. Felipe Souza, Dr. Qinglong Yan, Dr. Karina Hermawan, Dr. Jun Hyeong Park, Dr. Daisik Nam, Dr. Rezwana Rafiq, and Chenying Qin for their guidance in terms of both academic and personal life. Graduate school is hard and serious. But all my ITS friends and colleagues made my graduate life cheerful and colorful! So, I would like to say thank you to Dr. Xuting Wang, Lu Xu, Dingtong Yang, Youngeun Bae, Sunghi An, Irene Martinez, Sina Babbagh, and David Sedath, etc.

I also gratefully appreciate the support provided by the California Department of Transportation District 11 Electrician, Thomas Shepard for his support of data collection.

Last but not least, I would like to thank my family especially my mom for her love, care, support, and patience. She helped me get through the darkest days during my graduate school. She has always trusted me even though I felt hopeless myself. I feel tremendously grateful and lucky to have a such great mother.

# CURRICULUM VITA

Yiqiao Li

## EDUCATION

### **Ph.D., Civil and Environmental Engineering, 2021**

#### **University of California, Irvine**

Dissertation: Resilient Spatiotemporal Truck Monitoring Framework using Inductive Signature and 3D Point Cloud-based Technologies

Committee members: Stephen G. Ritchie (Chair), Amelia C. Regan, Wilfred W. Recker

### **M.S., Civil and Environmental Engineering, 2017**

#### **University of California, Irvine**

Thesis: Investigation of a New Method for Improving Individual Vehicle Speeds Estimation with Advanced Loop Data

Committee members: Stephen G. Ritchie (Chair), Wilfred W. Recker, R. Jayakrishnan

### **B.S., Civil and Environmental Engineering, 2015**

#### **Beijing University of Technology**

Thesis: Analysis on Pedestrian Level of Service in Metro Transfer Corridor based on the Physiological Characteristics using Physiological Sensors

Advisor: Lishan Sun

## ACADEMIC AWARDS

Transportation Forecasting Competition, Contest Winner, TRB	2019
TRB Travel Time, Speed, and Reliability (TTSR) Subcommittee, the Best Paper Award	2019
Associated Graduates Students in University of California, Irvine, Travel Grant	2018
Outstanding Graduates Awards in Beijing University of Technology	2015
The First Prize Scholarship in Beijing University of Technology	2012, 2014

## RESEARCH EXPERIENCE

### **Graduate Student Assistant, UC Irvine Institute of Transportation Studies**

2015 – 2021

- Enhanced Truck Counts for California (Caltrans)
- Network-wide Truck Tracking and Weight Estimation (SB1)

- Improved California Truck Traffic Census Reporting and Spatial Activity Measurement (SB1)
- Investigation of Truck Data Collection using LiDAR Sensing Technology along Rural Highways (PSR)

## **TEACHING EXPERIENCE**

**Teaching Assistant, Civil and Environmental Engineering, University of California, Irvine, Spring 2021**

CEE 124 Transportation System IV: Freeway Operation and Control

## **PUBLICATIONS**

### **Journal Publications**

**Li, Y.\***, Tok, A. Y., & Ritchie, S. G. (2019). Individual truck speed estimation from advanced single inductive loops. *Transportation research record*, 2673(5), 272-284. (Best Paper Award)

### **Conference Publications**

**Li, Y.\***, Allu, K., Sun, Z., Tok, A., Feng, G, Ritchie, S. G. (2021). An Ensemble Approach to Truck Body Type Classification using Deep Representation Learning on 3D Point Sets. *Transportation Research Board 100<sup>th</sup> Annual Meeting*.

**Li, Y.\***, Tok, A., Ritchie, S. G. (2021). A Deep Ensemble Neural Network Approach for FHWA Axle-based Vehicle Classification using Advanced Single Inductive Loops. *Transportation Research Board 100<sup>th</sup> Annual Meeting*.

Yan, Q., **Li, Y.**, Tok, A., & Jin, W. L.\* (2019). Calibration of Unifiable Multi-lane Multi-class Fundamental Diagrams Using Inductive Signature Technologies. *Transportation Research Board 98<sup>th</sup> Annual Meeting*, No. 19-05997.

Luo, W., Sun, L.\*, **Li, Y.**, Zhang, Y., & Rong, J. (2018, November). A New Attempt for Pedestrian Level of Service in Subway Transfer Corridor Based on The Physiological Characteristics. In the *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1939-1943). IEEE.

### **Presentations**

**Yiqiao Li\***, Andre Tok, Stephen G. Ritchie, Enhancing truck activity monitoring through the integration of Bluetooth and Inductive loop Signature data. Transportation Research Board Freight and Marine “Work-in-Progress” Lightning Talks and Poster Session (TRB2020), Washington D.C., Virginia, USA



**Yiqiao Li\***, Andre Tok, Stephen G. Ritchie, Enhancing Truck Activity Monitoring System by using Advanced Traffic Sensors. Presentation of 2019 Transportation Research Board Innovations in Freight Data Workshop, Irvine, California, USA

**Yiqiao Li\***, Andre Tok, Stephen G. Ritchie, From Advanced Single Inductive Loop Data to Individual Truck Speed Estimation – A Machine Learning Approach. Presentation of 2019 Irvine Symposium on Emerging Research in Transportation (ISERT), Irvine, California, USA

**Yiqiao Li\***, Andre Tok, Stephen G. Ritchie, Individual Truck Speed Estimation from Inductive Single Loops. Presentation of 2019 Transportation Research Board Annual Meeting, Washington D.C., Virginia, USA

**Yiqiao Li\***, Andre Tok, Stephen G. Ritchie, Improved truck activity measurement through the fusion of advanced point detector and wireless sensor network, 2018. Transportation Research Board Freight and Marine “Work-in-Progress” Lightning Talks and Poster Session (TRB2018), Washington D.C., Virginia, USA

**Yiqiao Li\***, Andre Tok, David Sedath, Stephen G. Ritchie, 2018. Spatiotemporal Truck Network Flow Estimation through the Fusion of Inductive Loop Signature and Bluetooth Data. Presentation at National Travel Monitoring Exposition and Conference (NaTMEC2018), Irvine, California, USA

David Sedath\*, **Yiqiao Li**, Andre Tok, Stephen G. Ritchie, 2018. Intersection Pedestrian Tracking through Bluetooth Technology. Presentation at National Travel Monitoring Exposition and Conference (NaTMEC2018), Irvine, California, USA

**Yiqiao Li**, 2014. Investigation of vehicles detection system through the use of Internet of Things (IoT) sensors, 2014 Competition of Transport Science and Technology for Students in Beijing University of Technology

## **PROFESSIONAL ACTIVITIES & SERVICE**

### **Reviewer:**

- IEEE Intelligent Transportation Systems Society Conference (ITSC 2018)
- Transportation Research Board Annual Meeting Proceedings (2020, 2021)

### **TRB Committee Friends:**

- Standing Committee on Highway Traffic Monitoring (ACP70)
- Standing Committee on Freight Transportation Data (AED70)

## **RESEARCH INTERESTS**

Traffic surveillance using Advanced Traffic Sensors (Inductive Loop detector, Bluetooth, LiDAR, Camera), Transportation Big Data Analytics, Apply Artificial Intelligence and Computer Vision on Transportation Applications

## **SKILLS**

### **Programming Language:**

- Proficient in Python, R
- Experiences with Java, SQL, Bash, C#, Matlab

**Machine Learning Framework:** Keras, Tensorflow, Pytorch

**Operating Systems:** Linux (Distro: Ubuntu, Raspbian), Windows

**Documentation:** LaTeX, Markdown, Microsoft Office

**Optimization Solver:** Gurobi

**Simulation Software:** TransModeler, TransCAD, Synchro

**GIS Application:** QGIS

## **ABSTRACT OF THE DISSERTATION**

Resilient Spatiotemporal Truck Monitoring System using  
Inductive Signature and 3D Point Cloud-based Technologies

by

Yiqiao Li

Doctor of Philosophy in Civil and Environmental Engineering

University of California, Irvine, 2021

Professor Stephen G. Ritchie, Chair

Understanding the spatiotemporal distribution of commercial vehicles is essential for facilitating strategic pavement design, freight planning, and policy making. Hence, analysts and researchers have been increasingly interested in collecting more diverse high granularity truck data across different truck characterization schemes to meet these various needs across the roadway network to better understand their distinct operational characteristics and dissimilar impacts on infrastructure and the environment. Existing truck monitoring infrastructure is limited in spatial coverage across the roadway network due to high installation and maintenance costs. The recently developed Truck Activity Monitoring System (TAMS) by the University of California Irvine Institute of Transportation Studies provides a cost-effective solution for monitoring truck movements statewide across California along major freeways networks through existing inductive loop infrastructure enhanced with inductive signature technology. Nonetheless, it possesses three major limitations: model bias against underrepresented truck classes, spatial coverage limitation on rural highways, and system obsolescence over time.

This dissertation explored a resilient spatiotemporal truck monitoring system using inductive loop signature and multi-array Light Detection and Ranging (LiDAR) sensor technologies to address the aforementioned issues and to improve truck monitoring capabilities across the roadway network. The designed system comprises three major parts: Inductive loop sensors for major highway truck monitoring, multi-array LiDAR sensors for rural highway truck monitoring, and a self-learning truck classification framework through a sensor integration framework.

The first part of the system was built upon the existing Truck Activity Monitoring System (TAMS) developed by ITS Irvine and addresses prediction model biasness caused by inherently imbalanced truck datasets to provide reliable truck speed estimation and truck classification data.

The second part explored non-intrusive LiDAR-based sensing technologies to fill the surveillance gap along rural highway corridors. This section developed a truck classification method using a LiDAR sensor oriented to provide a wide field-of-view of roadways.

Finally, a self-learning framework for truck classification systems was designed to address system obsolescence through the integration of inductive loop sensors and LiDAR sensors, the latter of which has been proven in this dissertation to have the ability to recognize truck axle configuration. This framework enhances the resilience of the signature-based FHWA classification model with an intelligent system update to accommodate the change of the truck designations over time and significantly reduces the overall burden of periodic model calibration by utilizing the information stored in the legacy model.

# CHAPTER 1 : Introduction

## 1.1 Impacts of Trucks

Trucks hauled 10.4 trillion dollars of domestic freight in 2017 and transported the highest percentage of freight in value (89 percent), tonnage (76 percent), and ton-miles (55 percent) among single modes (*Commodity Flow Survey, 2017*).

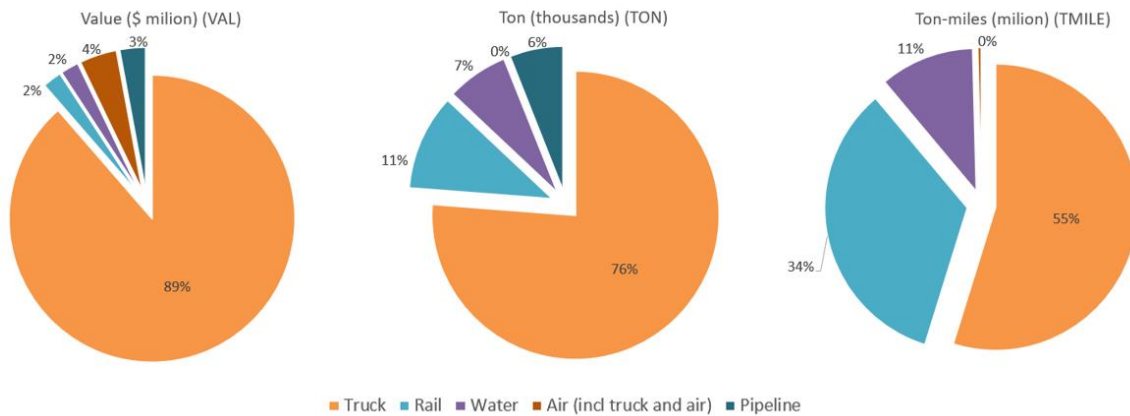


Figure 1.1 Commodity Values, Tons and Ton-miles transported by Trucks (U.S. Department of Transportation et al., 2017)

According to forecasts from the Freight Analysis Framework, the value of freight transported by trucks is expected to follow a rising trend. The per-ton values moved by trucks have been forecast to reach 18,735 dollars in 2045. The growing demand for highway freight transportation derives trucking-related concerns. These concerns come from the severe adverse impacts attributed by truck across different classification scheme on roadway safety (Dong et al., 2015), pavement design (Gillespie, T.D, Karamihas, S.M. & Sayer, 1993), as well as a greenhouse gas and pollutant emissions (Guensler et al., 2005).

**Roadway Safety:** Commercial trucks account for 8 percent of U.S. highway traffic volume.

Yet, they are disproportionately involved in 11 percent of fatal crashes and responsible for 4,500 deaths per year in the U.S. alone. Previous research shows that an increase in truck population on the highway results in a magnified increase in the probability of a severe crash (Dong et al., 2015). Moreover, the contribution to the crash fatalities varies by location (e.g. rural, urban, work zone) and truck configuration (U.S. Department of Transportation, 2019). However, the roadway safety analysis concerning different truck types is a lack in comprehensive research due to the limited resolution of current truck data.

**Pavement Structural Damage:** The American Association of State Highway and Transportation Officials (AASHTO) introduced the concept of 18,000 lbf (pound force) Equivalent Single Axle Loads (ESALs) to evaluate the impact of truck axle configuration on the pavement at various locations (AASHTO, 1993), based on a large-scale road test in the 1960s. The reference unit of ESAL represents a level of damage that an 18,000 lbf single axle load would have on the pavement. As shown in Figure 1.2, the ESAL values of passenger vehicles are negligible and approximately equal to zero. Compared to two-axle passenger vehicles, trucks produce disproportionately negative impacts on pavement structures, whereas different truck classes bring a varying degree of damage to various types of roadway facilities.

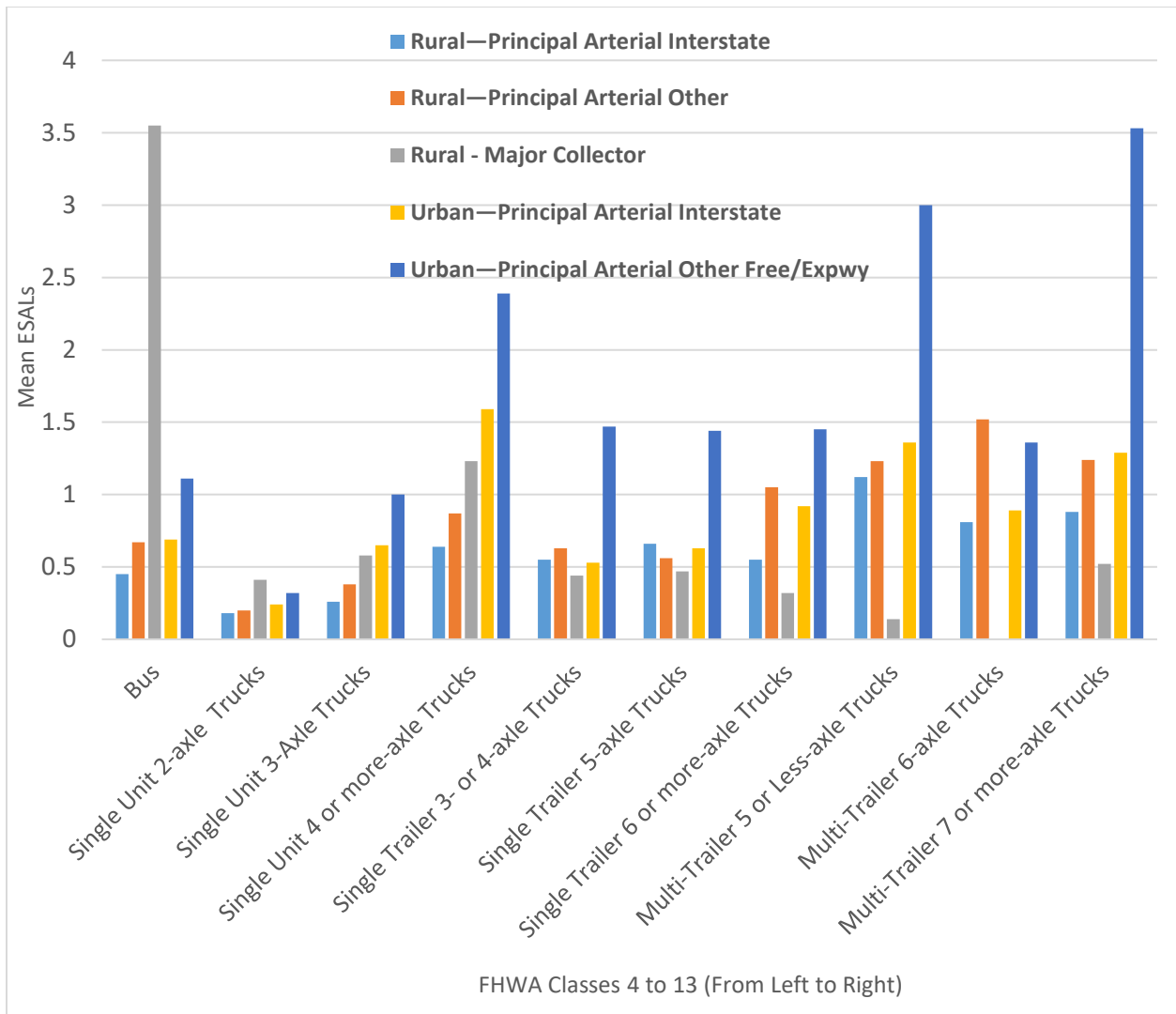


Figure 1.2 Average ESALs by FHWA Vehicle Classes and Infrastructure Functional Classes (Note: Since Motorcycles, passage cars, and SUV/Pickup trucks do not significantly contribute to the ESALs, they are negligible and their ESALs are approximately 0.) (Federal Highway Administration, 2013; Schmoyer & Hu, 1998)

Many transportation agencies rely on Weigh-In-Motion and Automatic Vehicle Classification sites to collect axle-based vehicle classification counts for the use in pavement design. However, the spatial coverage of these detection sites across the highway network is limited due to high installation and maintenance costs. Therefore, reliable truck data sources with extensive spatiotemporal coverage are required to evaluate the pavement design of truck

corridors in response to the potential damage caused by various compositions of truck classes.

**Environmental Impact:** Commercial trucks are the main sources of air pollution on roadway networks, even though they only account for a small proportion of U.S. highway traffic. As illustrated in Figure 1.3, trucks, particularly heavy-duty trucks, contribute over 50 percent of the pollutants, such as carbon monoxide (CO), hydrocarbons (HC), nitrogen oxides (NOx), and particulate matter with a diameter of 2.5 micrometers or smaller (PM2.5).

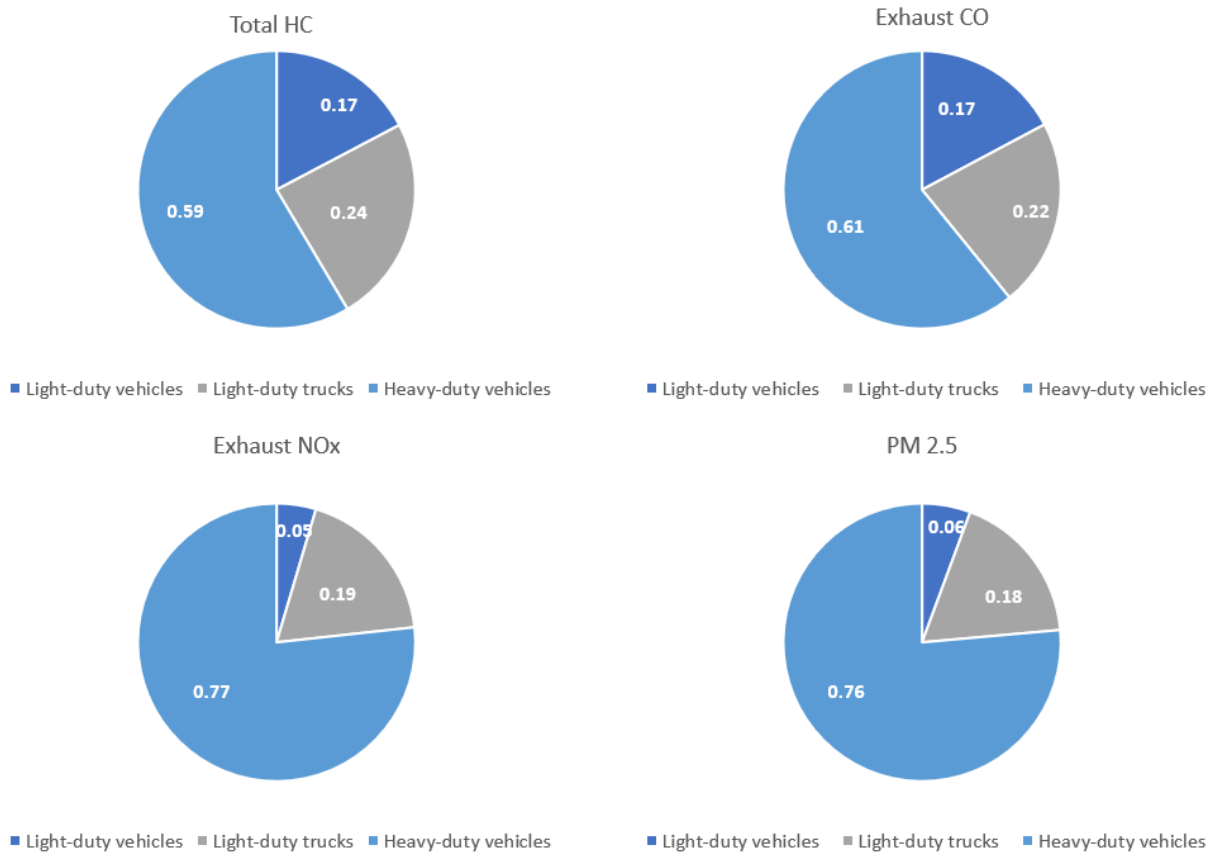


Figure 1.3 Percentage Estimated U.S. Average Vehicle Emission Rate per Vehicle by Vehicle Type using Gasoline and Diesel (Statistics Bureau of Transportation, 2021) Note: Light-duty vehicles: passenger cars. Light-duty trucks: trucks with two axles and four tires. Heavy-duty vehicles: trucks with more than two axles or four tires



In summary, analysts and researchers have been increasingly interested in collecting more diverse high-granularity truck data across different truck characterization schemes to meet these various needs across the roadway network to better understand their distinct operational characteristics and dissimilar impacts on infrastructure and the environment.

## 1.2 Current Truck Data Sources and Concerns

A comprehensive understanding of truck activities is required to address the adverse impact caused by trucks. Data sources, which are generally used for freight analysis, are summarized below as three primary parts: survey data, operation data source, and emerging technologies (Table 1.1).

Table 1.1 Current Truck Data Sources

<b>Survey Data</b>	<b>Operation Data Source</b>	<b>Emerging Technologies</b>
Commodity Flow Survey	Truck GPS	Truck Activity Monitoring System
Freight Analysis Framework	Weigh-In-Motion	LiDAR data
Vehicle Inventory and Use Survey	ATR Classification Count Data	Bluetooth data

Data from such current detection technologies as truck Global Position System (GPS) data have been adopted for freight transportation planning purposes. Such data have been used in the development of truck tour-based models (Sharman & Roorda, 2011; Zanjani et al., 2015), the identification of trip destinations (Sharman & Roorda, 2011), and the estimation of such truck performance measures as travel time, travel speed and travel distance (X. Ma et al., 2011; Wang et al., 2016). However, due to confidentiality, truck GPS data provide details of the truck characteristics on a limited level. For example, the truck GPS data from the American Transportation Research Institute (ATRI) contain only truck type information on an aggregated level (e.g., heavy trucks and light trucks) (Zanjani, 2014). On the other

hand, such infrastructure-based systems as Automatic Traffic Recorder (ATR) classification count data and Weigh-in-Motion (WIM) data can provide truck classification information. In particular, WIM devices are designed to capture and record gross vehicle weights (GVW), axle weights, axle spacing, and axle-based vehicle classification (FHWA 13 Vehicle Category Classification) (Federal Highway Administration, 2013) as vehicles traverse the WIM sensors. However, WIM stations are sparsely deployed across the network due to high construction and maintenance costs. Furthermore, although axle-based vehicle classification data are essential for estimating pavement damage caused by trucks with different axle configurations, such datasets provide limited information related to a truck's commercial purpose, which is an important input to highway freight transportation planning models (Beagan et al., 2019; Schaefer, Ron; Worth, Monica; Heilman, Jonathan; Kehoe, 2017). U.S. Census Bureau data sources, such as the Commodity Flow Survey (CFS), Services Annual Survey, and Vehicle Inventory and Use Survey (VIUS) have been conducted in an attempt to fill this gap and meet the needs of planners and practitioners. Typically, such survey data as the VIUS tie truck body types to their commodities hauled, where the commodities hauled can be inferred from the available truck body type classification data. Hence, these types of data can provide the commercial and operational characteristics of the Nation's truck population and support the analysis of environmental impacts of traffic emissions as well as estimating fuel demand. However, survey data are static in nature. It is hard for survey data to capture the dynamics of truck activities on both spatial and temporal dimensions. Additionally, a new publicly available data source – the Truck Activity Monitoring System (TAMS) (Tok et al., 2017)—has been developed by ITS-Irvine. This high-resolution data source, which leverages existing advanced loop sensor infrastructure, can provide truck data

with detailed truck body types (Hernandez, 2014). Nonetheless, this truck activity monitoring system currently experiences three limitations: model bias against under-represented truck classes, spatiotemporal coverage limitation on rural highways, and model obsolescence over time. First, the truck class distribution for classification schemes such as the FHWA axle-based classification scheme is imbalanced in nature. The existing signature-based FHWA classification model overlooks model performance on minority classes, even though many of them may pose disproportionately adverse impacts on infrastructure and the environment. Therefore, previous models have been unable to adequately classify under-represented classes, and the overall performance of the models is often masked by excellent classification accuracy of majority classes, such as passenger vehicles and five-axle tractor-trailers. Second, although inductive loop sensors are widely deployed in many urban and interstate highway corridors in the U.S., their coverage remains limited along rural highway corridors that also contribute significantly to the economy. However, the installation of such an intrusive sensor requires pavement intrusive installations and lane closures that are both cost- and labor-intensive. Therefore, it is impractical to implement pavement intrusive sensors extensively along the rural highway network which results in a truck monitoring gap in rural areas for long-term operations. Third, inductive loop sensor systems have the ability to provide truck classifications of comparable accuracy to current axle-based sensor systems when they are updated with inductive signature technology and advanced machine learning models. However, the existing truck population is expected to transition over time and be replaced with newer models that may possess distinct and different inductive signature characteristics. Consequently, the performance of inductive signature-based systems may not be optimal in classifying newer trucks operating on the highway over time.

### **1.3 Proposed Solution Overview**

This dissertation describes the design and development of a resilient spatiotemporal truck monitoring framework to fill the aforementioned gaps in the current system. This new framework comprises three major components to address the corresponding concerns: inductive loop sensors for major freeway truck monitoring, multi-array LiDAR sensor for rural highway truck monitoring, and a self-learning framework through the integration of inductive loops and LiDAR sensors. The first part of this dissertation develops an individual truck speed estimation model as well as a signature-based FHWA classification model with the focus on addressing the low performance of minority classes. The truck speed estimation model is developed on a truck-focused dataset and truck speed-related features were extracted to enhance the model performance on trucks. In addition, a bootstrap aggregation deep neural network model is proposed to classify vehicles on the basis of the FHWA-CA scheme using inductive loop signature data. This method attempts to remedy the imbalanced dataset issue in the process of developing prediction models on the algorithm level to provide reliable FHWA vehicle classification data. The second part of this dissertation is focused on filling the truck monitoring gap on rural highways using a mobile sensor solution—a multi-array LiDAR sensor. First, a new vehicle reconstruction framework with ground plane consideration is designed to enrich the sparse point cloud obtained from each LiDAR frame. Second, the lower profile of the truck is extracted from the reconstructed truck point cloud and used as input to an ensemble deep neural network model to classify trucks according to the FHWA classification scheme. Finally, the PointNet deep representation learning algorithm is utilized to classify trucks based on their detailed truck body types. The

final part of this dissertation describes the design of a self-learning framework through the integration of inductive loops and LiDAR sensors. The LiDAR-based FHWA vehicle classification model serves as an automated labeling platform to generate class labels to validate and calibrate the signature-based vehicle classification model without human involvement. This framework enhances the resilience of the signature-based FHWA classification model and significantly reduces the overall burden of periodic model calibration.

### 1.4 Dissertation Outline

The structure of this dissertation is described in Figure 1.4.

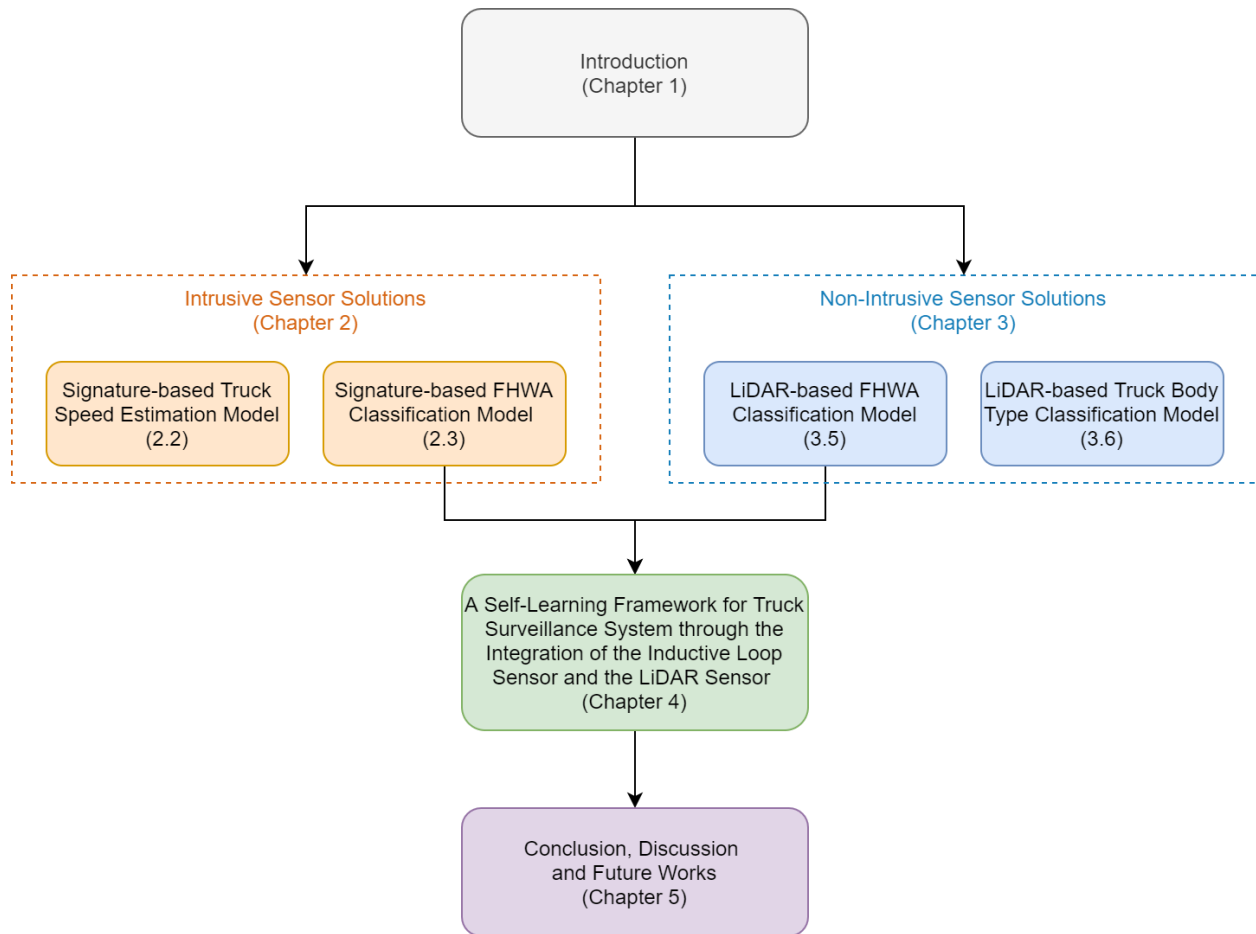


Figure 1.4 Relationship between Each Chapter

The rest of this dissertation is organized as follows. Chapter 2 presents the enhancement of the truck monitoring framework on major highways using inductive loop signature data. This chapter addresses model bias issues on both the data and algorithm level. Chapter 2 presents the development of an individual truck speed estimation model with a truck-focused dataset using inductive loop signature data and subsequently demonstrates a bootstrap aggregation deep neural network model to resolve the challenge of imbalanced datasets in the FHWA vehicle classification problem.

In Chapter 3, a new truck classification method is developed using a non-intrusive sensor – multi-array 3D LiDAR – that can be used to improve truck monitoring on the rural highway network. First, a new vehicle point cloud reconstruction framework with ground plane consideration is developed to merge multiple consecutive frames associated with each vehicle object to provide its dense point cloud representation. Next, the reconstructed point cloud is used to develop an FHWA vehicle classification model and a truck body type classification model using a classic deep neural network model architecture and a representation learning algorithm, respectively.

In Chapter 4, a self-learning vehicle classification framework is designed through the integration of inductive loops and LiDAR sensors to address the system obsolescence issue as a result of truck population turnover. The LiDAR-based FHWA vehicle classification model developed in Chapter 3 serves as a data labeling platform to validate and calibrate the signature-based vehicle classification model proposed in Chapter 2. An adaptive transfer learning framework is adopted to enhance the overall model performance on both the archived statewide dataset and the newly collected dataset without compromising computation efficiency.

Chapter 5 presents the conclusion of this dissertation with a discussion of potential future research directions.

## **CHAPTER 2 : Intrusive-Sensor Solutions: Major Highway Truck Monitoring using Advanced Inductive Loops**

### **2.1 Individual Truck Speed Estimation from Advanced Single Inductive Loops**

#### **2.1.1 Introduction**

Truck speed has been considered an important traffic estimate that can be helpful for identifying congestion (Cambridge Systematics Inc. & Battelle Memorial Institute, 2005), evaluating roadway safety (Malyskhina & Mannering, 2008), and estimating emissions (United States Environmental Protection Agency, n.d.). Researchers have been attempting to improve the accuracy of truck speed estimation through the use of advanced traffic technologies. Traffic technologies used for speed estimation can be categorized as intrusive and non-intrusive. Non-intrusive detection technologies can provide relatively accurate measurements of aggregated vehicle speed and travel time. However, the spatial coverage of these technologies is limited. For example, the penetration rates of Bluetooth devices vary spatially and temporally (Vo, 2011). Therefore, speed estimation models developed by using non-intrusive sensors such as Bluetooth and Wi-Fi are not effective at locations where the traffic streams do not contain vehicles equipped with probe devices, such as in rural areas (Thiagarajan et al., 2014). Furthermore, it is hard to identify whether detected Bluetooth and Wi-Fi devices are associated with trucks or passenger vehicles. Compared to Bluetooth and Wi-Fi sensors, probe devices like truck Global Positioning System (GPS) provide better spatial coverage (Zanjani, 2014). However, sample truck GPS data are not necessarily representative of the population of all vehicles (Zanjani et al., 2015). Since different types of vehicles possess different speed characteristics (Meher et al., 2013), information from biased sampling may lead to a biased speed representation of total truck and traffic flow.



Conversely, information collected from conventional inductive loop detectors provides excellent temporal coverage since data can be collected continuously from these sensors. Single-loop detectors are widely installed in the US and are commonly used for freeway traffic surveillance. However, a conventional single inductive loop sensor provides only duration measurements at the vehicle level, which is both a function of vehicle speed and length. Hence, aggregated speed estimates can be obtained from conventional single inductive loop sensors if the average vehicle length of the traffic stream is known. This is typically done by assuming the effective vehicle length of vehicles traversing the sensor over a pre-defined time period. The effective vehicle length basically represents the sum of the single inductive loop sensor and vehicle length. However, the distribution of vehicle lengths can vary dynamically in a traffic stream and across lanes. Hence, the conventional g factor method (Athol, 1965), which considers the effective vehicle length as a constant, may result in significant errors. The moving median method (Coifman et al., 2003), sequence method (Coifman & Kim, 2009), and distribution method (Coifman & Kim, 2009) were subsequently developed to improve the estimation accuracy of average speeds under heterogeneous traffic conditions. Nonetheless, these models still show limitations in estimating the speed of traffic streams accurately when a large percentage of long vehicles is present (Coifman & Neelisetty, 2014). The new sequence method and the hybrid method proposed by Coifman and Neelisetty (2014) improved estimation accuracy under high truck flows but still yielded 5.6 percent average absolute percentage error at the 30-second aggregation level under free-flow conditions. Although the hybrid model achieved better performance compared with previous studies, it was unable to provide accurate truck speed estimation. Hence, the

challenge remains for conventional single inductive loop detectors to provide accurate truck speed estimation.

With advanced detector cards, the inductive loop detector system can provide much more detailed information through inductive waveforms (or signatures) generated by each individual vehicle. The potential for estimating vehicle speeds using inductive signature data was initially investigated by Sun and Ritchie (1999), and subsequently explored by others to further improve speed estimates (Oh et al., 2002; Park & Ritchie, 2010; Tok et al., 2009). However, a common weakness for truck speed estimation was that the datasets used contained only a limited proportion of trucks. Hence, the performance of these models has not been well-tested under traffic conditions with high truck proportions. In this chapter, a recent signature-based speed estimation model was tested on a truck-focused dataset and revealed a significant reduction in model accuracy.

This research aims to fill the gap and improve the accuracy of individual truck speed estimation. First, compared to the simple moving average method adopted by previous research (Tok et al., 2009), a progressive moving average method preserves the old datum points at the leading edge of a signature even with a larger averaging window size. Subsequently, the optimal window size and the optimal cutting threshold are analyzed in this chapter. Then, a new speed-related feature was investigated and tested. Next, based on the new speed-related feature, all vehicle observations are clustered into groups by using a modified feature-weighted K-means algorithm. Finally, a new multi-layer perceptron neural network model is constructed to classify single loop signatures into pre-determined clusters. An accurate truck-focused individual vehicle speed estimation (ATISE) model is developed

and applied on both a truck-focused dataset as well as a general dataset comprising mostly non-trucks. Previous models have been applied only on general datasets and reported with the overall mean absolute error and the overall mean absolute percentage error. Since trucks only account for a small portion of most datasets, the average accuracy can still be high even if the truck speed estimates have significant errors. Hence, the deficiencies of previous models in estimating truck speeds may not be apparent. To address previous dataset limitations, we develop and test our model on a dataset comprising 80 percent trucks. The new model showed a significant improvement over the best previously developed signature-based individual vehicle speed estimation (ISE) model. The overview of the ATISE model structure is shown in Figure 2.1.

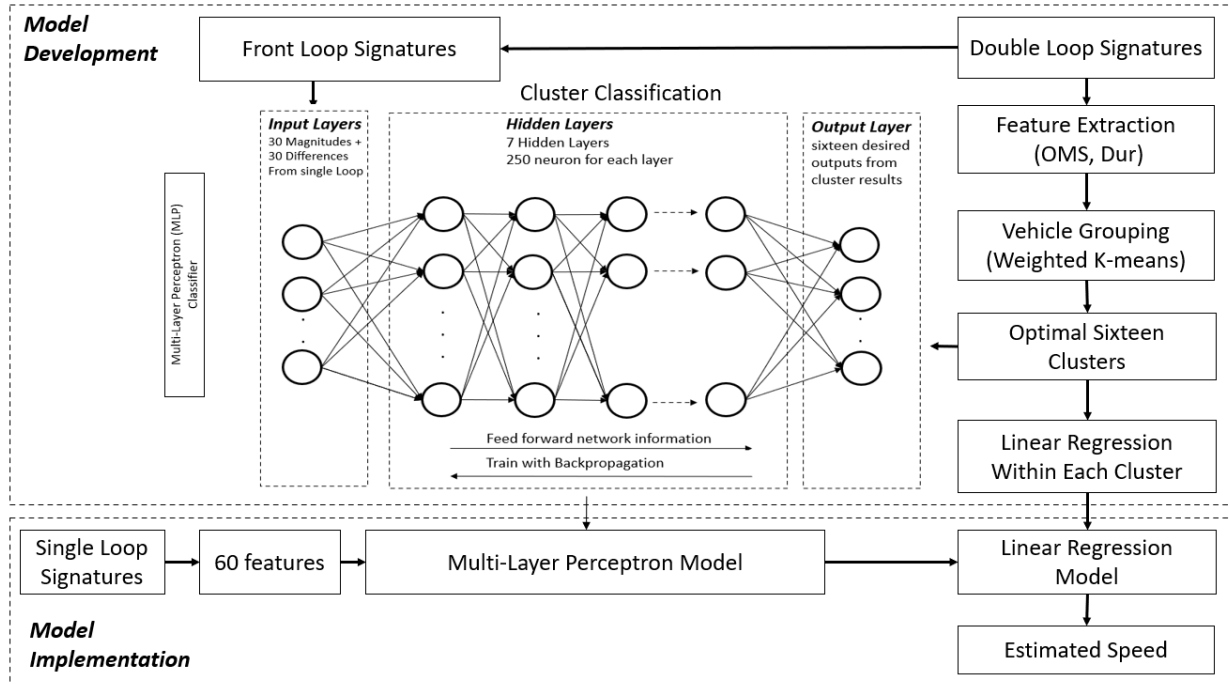


Figure 2.1 Overview of the AISE Model

## 2.1.2 Literature Review

### 2.1.2.1 Conventional Loop Methods

Most conventional loop speed estimation methods are based on the vehicle length and duration relationship discussed in the previous section. The first-speed estimation method that utilized data obtained from single inductive loop detectors was proposed by Athol (1965). He introduced the  $g$  factor into speed estimation ( $v = \frac{N}{g \times T \times Occ}$ ). Here,  $v$  represents space mean speed.  $T$  indicates the observed time interval.  $Occ$  represents the percentage of time that the loop sensor is occupied by vehicles within each interval.  $N$  denotes the number of vehicles within each time interval. The  $g$  factor is an estimator that incorporates site characteristics of effective vehicle length, and hence varies by site. However, effective vehicle lengths vary significantly in both the spatial and temporal domain. In response, many

researchers have sought improved methods to address the bias caused by dynamic vehicle mix and congestion effects. Pushkar et al (Pushkar & Acha-daza, 1994) first estimated average vehicle lengths based on cusp catastrophe theory, then developed a new speed model based on estimated average vehicle lengths. Their speed model showed improved performance over the use of constant average vehicle lengths. Rather than using a fixed pattern of effective vehicle lengths for different time periods within one day, Coifman (2001) used an exponential filter to dynamically update the average vehicle length, which improved the reliability of his speed model. Subsequently, Coifman et al (2003) found that the use of median speed was much less sensitive to outliers compared to the use of mean speed. However, this model required observing several vehicles in a given sample duration to reduce the impact of long vehicles and was not advisable to apply on datasets with low volumes (Coifman et al., 2003). Then, Coifman and Kim (2009) refined the existing speed estimation methods, the sequence (Coifman & Neelisetty, 2004) and moving median method (Coifman et al., 2003), and proposed a distribution method that used an on-time distribution to estimate vehicle on-times, in order to address the weakness of the sequence method under congested conditions. The transferability of the model was not validated. Later, Lao et al analyzed the on-time distribution pattern of vehicles and used a Gaussian Mixture Model (GMM) to classify different types of vehicles by their lengths (Lao et al., 2012). They iteratively updated the classified vehicle volume estimates and aggregated speeds of the traffic streams. Compared to previous methods (Coifman & Neelisetty, 2004) (Coifman et al., 2003), Lao's model provided improved speed estimation across a variety of traffic conditions and addressed transferability issues. However, the authors remarked that the model needs to be calibrated if long vehicles are dominant in the traffic stream. In 2014, Coifman and

Neelisetty (2014) presented a new sequence method to improve speed estimation under severely heterogeneous traffic conditions. They deduced individual vehicle length by observing the extreme on-time ratios and extracted two types of sequences, a long vehicle followed by a short vehicle and a short vehicle followed by a long vehicle. Since the sequence method relied on the passage of sequences and the location, a hybrid method was proposed to reduce the estimation bias caused by different sequence occurrences. The model started by detecting the presence of sequences in the sample time period. If there was a sequence in the sample, the median speed of all vehicles in the sequence during the sampling time period was the estimated speed. Otherwise, if the sampling period was less than 2 minutes, the estimated speed was equal to the median speed of all vehicles in the sequence during the previous 2 minutes. Finally, they had to identify the free flow condition, with the assumption that the free flow speed was fixed. The model yielded a mean absolute error of 5.6% under 40% truck flows in free flow conditions. However, model performance was not reported under congested conditions with high truck proportions.

#### **2.1.2.2 Speed Estimation with Non-Intrusive Sensors**

In recent years, various sensors have been used in traffic surveillance. Detection technologies can be broadly classified as intrusive and non-intrusive. Pneumatic road tubes, inductive loop detectors, piezoelectric sensors, magnetometer-based sensors, and Weigh-in-Motion that involve in-pavement installations are examples of intrusive detectors. On the other hand, video image processing, microwave radar, infrared sensors, GPS, Bluetooth sensors, and probe vehicles that are installed out-of-pavement can be classified as non-intrusive detectors. Non-intrusive detection technologies such as GPS, Bluetooth sensor, and probe vehicles are currently used for performance measurement of roadway networks. Since

vehicles equipped with a GPS device can report the individual vehicle speed, Cheu et al (2001) calculated the optimal sample size of probe vehicles equipped with GPS devices and obtained relatively accurate speed estimates along arterial links. Unfortunately, the sample GPS data are only a small portion of overall traffic and are not necessarily representative of the population of all vehicles (Zanjani et al., 2015). Different types of vehicles possess different speed characteristics (Mehar et al., 2013). Thus, sampling bias may adversely affect the speed representation of the truck population. Bachmann et al (2013) integrated Bluetooth data and conventional loop detector data by using a data fusion algorithm and improved aggregated speed estimation on freeways. However, such sensor technologies as Bluetooth and Wi-Fi are not designed to associate estimated vehicle speed with vehicle types. Therefore, these sensors inherently lack the ability to provide individual truck speed estimation.

### **2.1.2.3 Inductive Signature Method**

Generally, the conventional inductive loop detector card with sampling rates of 240 Hz (Coifman & Neelisetty, 2014) is only capable of collecting bivalent data. The output of the detector card is either “0” or “1”, which can directly produce occupancy and vehicle counts. A “1” output signal represents the presence of a vehicle. Typically, the duration can be calculated from the presence of a vehicle passing over the loop detector. However, advanced detector cards can capture the detailed inductance change when a vehicle traverses an inductive loop sensor. The inductance change produces a high-resolution waveform called an “inductive vehicle signature”. The sampling rate of the detector cards used in this study was 1000 samples per second, which yielded a temporal gap of 0.001seconds between each data point.

Inductive signature data collected from a single loop sensor can also be used for speed estimation. Some features extracted from vehicle signatures may be associated with specific characteristics of the vehicle. Sun and Ritchie (1999) found that the inductive signature slew rate of individual vehicles had a strong linear correlation with vehicle speed, and developed a linear model between individual vehicle speed and slew rate. After conducting several statistical tests of the model, they found that their model was statistically significant, and the data could be well explained by the model. However, the linear relationship between different vehicles is different. To address this problem, Oh et al. (2002) used a probabilistic neural network (PNN) to group vehicles by vehicle length and applied a linear regression model for each vehicle group to increase the accuracy of the model. Later, Park and Ritchie (2010) introduced a shape parameter into a vehicle grouping model in order to get more specific vehicle types within each group in the model. Since two different vehicles traversing an inductive loop sensor with the same velocity may have different slew rates and duration, Tok et al. (2009) adopted an approach that applied k-means clustering and artificial neural network classification methods to group vehicles into homogeneous groups based on natural slew and signature length characteristics. The linear model within each group was improved. However, the regression of the speed model corresponding to the cluster representing long vehicles was found to be significantly weaker than for smaller vehicles.

#### **2.1.2.4 Summary**

Generally, models developed using the inductive signature approach have better accuracy and transferability compared with models that use only conventional loop data. Compared to non-intrusive technologies, the inductive signature approach provides both a better temporal and spatial coverage. However, since there is a wide range of vehicle lengths within



the truck population, it is hard to provide reliable estimates for trucks. With these challenges, there has been limited research into truck-focused speed estimation. Unlike passenger vehicles, trucks have been found to generate distinct signature patterns across body types, and truck body types can be associated with different vehicle lengths. Hence, a well-structured model is needed to handle the diversity of vehicle lengths within the truck population. In this dissertation, we describe the development of a truck-focused individual vehicle speed estimation model through the use of advanced loop signature data to fill this information gap.

### **2.1.3 Data Description**

The data used in this section were obtained from two double inductive loop detector stations equipped with advanced loop detector cards. Both training and testing datasets were obtained from the inductive loop station along the I-680 freeway in the City of Walnut Creek, California near Oak Park Blvd. Data from three lanes in both directions were recorded at this station. In total, 21,127 vehicle signatures were collected at multiple periods over a one-month span and were pre-classified into five vehicle types developed by Hernandez (Hernandez, 2014). These comprise 4,156 passenger vehicles (20%), 11,691 single-unit trucks (55%), 3,078 semi-trailer trucks (14%), 329 multi trailer trucks (2%), and 1,873 single trailer trucks (9%). The second dataset used for validating the spatial and temporal transferability of the speed estimation model was obtained from the Patterson Pass double-loop station along the I-580 freeway near the city of Tracy, California. The signature data were collected on July 13<sup>th</sup> and July 14<sup>th</sup>, 2018, with 23 percent and 16 percent truck volumes, respectively. The spacing between the leading edges between each double inductive loop sensor was approximately 20 feet. The length of each individual loop sensor

was approximately six feet. The configuration of the double loop speed trap at both these sites provides the ability to obtain accurate individual vehicle speeds for developing and evaluating the single inductive loop speed estimation models.

## **2.1.4 Model Development**

### **2.1.4.1 Signal Processing**

Several steps were performed in the processing of raw inductive signatures generated by high-resolution inductive loop detector cards. First, dual inductive signature data were obtained from double loop inductive sensors to facilitate the measurement of vehicle speeds and the transformation of signature waveforms from the temporal to spatial domain to extract speed-invariant features. In this study, vehicle lengths less than 10 ft or larger than 85 ft were considered to be outliers and excluded before the signal processing step. Incomplete signatures which were caused by vehicle lane changing behaviors were also excluded. A progressive moving average (PMA) filter was subsequently used to filter out noise in the signature data. When the window size ( $ws$ ) was large, a simple moving average (SMA) will drop out the old datum points at the leading edge of a signature (Figure 2.2a) and may lead to an incomplete signature.

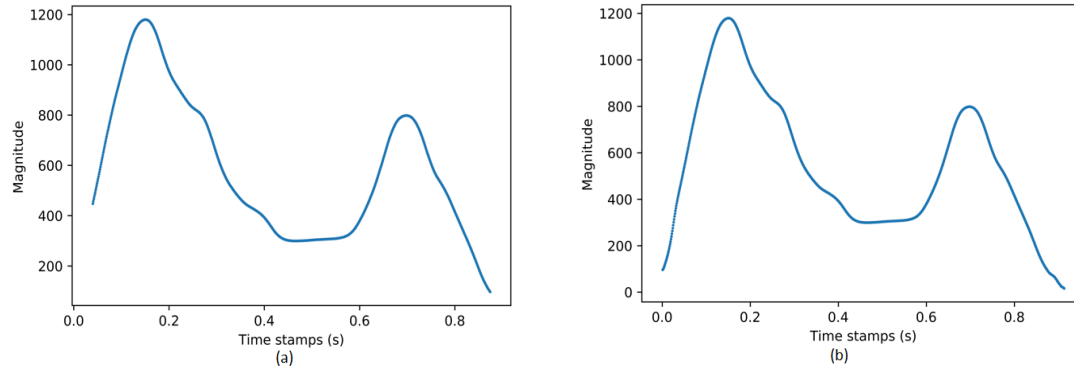


Figure 2.2 Comparison between simple moving average and progressive moving average (a) SMA with a window size of 79 (b) PMA with a window size of 79

To avoid this problem, we gradually increased  $ws$  until the maximum window size ( $mws$ ) was reached. The  $mws$  was subsequently maintained until the end of the signature where  $ws$  was gradually reduced. The  $ws$  of the progressive moving average method is a function of the index of each data point ( $i$ ). The relationship between  $ws$  and  $i$  is shown below, where  $n$  indicates the total number of data points in a signature, and  $mws$  is odd.

$ws(i) =$

$$\begin{cases} 2i - 1, & i \in \left[1, \frac{mws-1}{2}\right) \\ mws, & i \in \left[\frac{mws-1}{2}, n - \frac{mws-1}{2}\right] \\ 2(n - i) + 1, & i \in \left(n - \frac{mws-1}{2}, n\right] \end{cases} \quad (2.1)$$

Next, we calculated the estimated points with the dynamic  $ws$ . The estimated value for each index is shown below. Here,  $\bar{P}_{PMA_i}$  indicates each estimated value after applying the progressive moving average (PMA) filter.

$$\bar{P}_{PMA_i} = \begin{cases} \frac{x_1 + x_2 + \dots + x_{2i+1}}{WS(i)}, & i \in \left[1, \frac{mws-1}{2}\right) \\ \frac{x_{i-\frac{mws-1}{2}} + \dots + x_i + \dots + x_{i+\frac{mws-1}{2}}}{WS(i)}, & i \in \left[\frac{mws-1}{2}, n - \frac{mws-1}{2}\right] \\ \frac{x_{2i-n} + \dots + x_i + \dots + x_n}{WS(i)}, & i \in \left(n - \frac{mws-1}{2}, n\right] \end{cases} \quad (2.2)$$

Compared to the SMA used by previous research (Tok et al., 2009), the PMA method can prevent information loss on the leading edge of a signature, especially with the use of a larger window size (Figure 2.2b).

The optimal  $mws$  was obtained when the correlation between the overall maximum slew and true speeds obtained from double loops for the progressive filter was maximized. Through an exhaustive search on the interval [3, 99] the optimal  $mws$  was found to be 79. Next, inductive loop signatures were normalized against the maximum value on the magnitude domain to address inconsistencies in sensitivities across loop sensors. Finally, oscillations observed near the baseline inductance were eliminated by applying an optimal cut-off threshold. The purpose of the cut-off threshold is to exclude noise oscillations while preserving as much of the vehicle signature information as possible. Based on Figure 2.3, the correlation coefficient between overall maximum slew and true speeds was not found to be particularly sensitive to the cutting threshold.

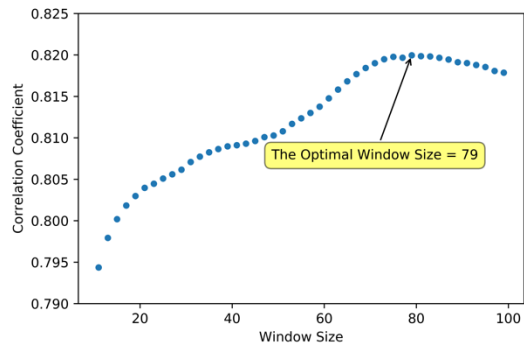


Figure 2.3 The Optimal Window Size

The cutting threshold was set to 0.1. The processed signature is shown in Figure 2.4.

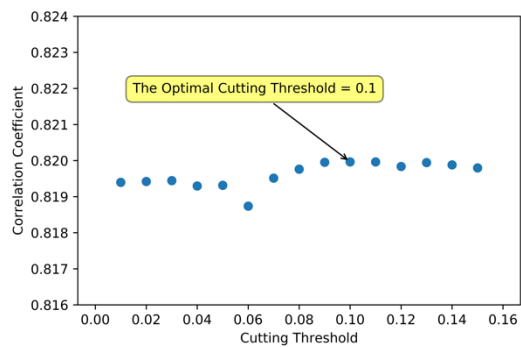


Figure 2.4 The Optimal Cutting Threshold

A raw signature and a processed signature are shown in Figure 2.5.

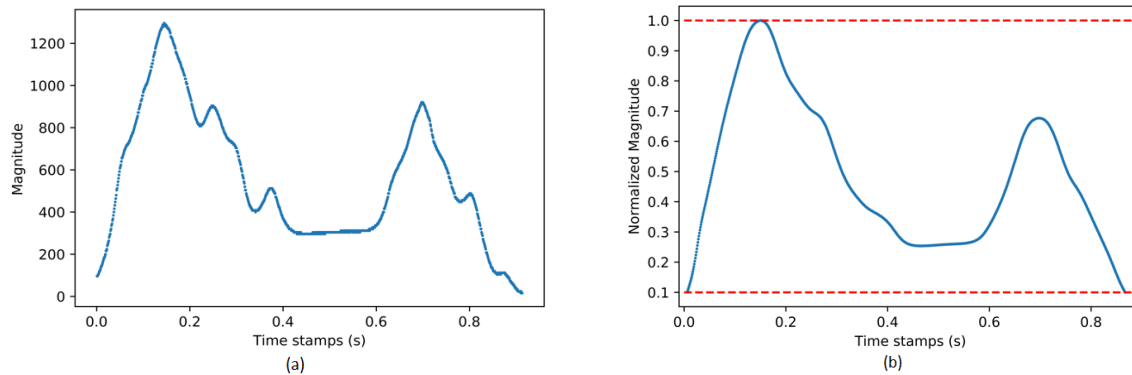


Figure 2.5 A Raw Signature and Processed Signature (a) A Raw Signature (b) A Processed Signature

### 2.2.4.2 Feature Extraction

A new speed-related feature was investigated in this study to improve the speed estimation of vehicles, particularly for trucks. Instead of the steepest slope on the leading edge of a vehicle signature—defined as Maximum Initial Slew (*MIS*) in this paper—we capture the maximum positive slope along the temporal domain of each signature and define this value as the Overall Maximum Slew (*OMS*). Figure 2.6 shows an example where the green slope on the first peak of the processed signature is less steep than the red slope on the second peak.

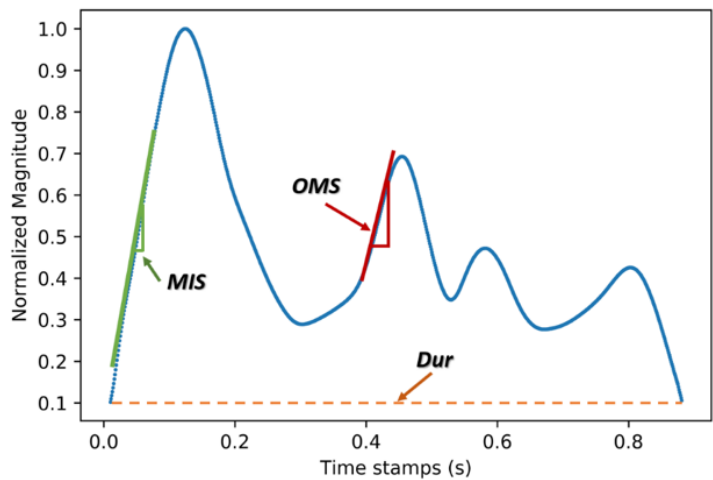


Figure 2.6 Feature Extraction

In this case, we choose the red slope value as the *OMS*. A comparison was performed on the correlation coefficient value between the *OMS* and the *MIS* to the actual speed value in a dataset comprising of 20 percent passenger vehicles and 80 percent trucks. The correlations between the *MIS* and *OMS* with true speed were found to be 0.80 and 0.82, respectively.

Duration (*Dur*) is the second feature adopted in the speed estimation model. The progressive moving average filter yields a complete signature pattern which is ideal for achieving a higher correlation with vehicle speeds.

#### 2.2.4.3 A Brief Introduction to the Speed Linear Regression Model

Instead of using *MIS*, this research adopted *OMS* as the explanatory variable in the speed estimation model.

$$v_{est} = \beta_0 + \beta_1 \cdot OMS + \beta_2 \cdot Dur^{-1} \quad (2.3)$$

Here,  $v_{est}$  indicates the estimated speeds. The  $\beta$ s are the model coefficients.

#### 2.2.4.4 Modified Weighted K-means Algorithm

Two different types of vehicles traveling at identical speeds traversing an inductive loop sensor may generate different *OMS*s and *Durs*, as these two features may possess different linear relationships with speeds across different vehicle groups. This leads to a low adjusted  $R^2$  if one linear model is estimated on the whole dataset (Sun & Ritchie, 1999). Thus, vehicles should ideally be grouped by their length-related features. The K-means clustering algorithm was applied to group similar vehicles based on those features. The objective of the K-means algorithm is to minimize the sum of Euclidean distances between each data point and its assigned cluster, which matches the goal that we want to achieve.

After proving that *OMS* is highly correlated with vehicle speeds, a temporal transformation should be applied to the extracted features (*OMS* and *Dur*). The mathematical explanation of the temporal transformation is as shown below:

$$NOMS = \frac{OMS}{v_{true}} \quad (2.4)$$

$$L_{sig} = v_{true} \times Dur \quad (2.5)$$

Here, *NOMS* represents the natural overall maximum slew.  $v_{true}$  represents the true speed obtained from double loop signatures.  $L_{sig}$  stands for the signature length and *Dur* is the duration measure. Prior to the clustering procedure, *NOMS* and  $L_{sig}$  are normalized against their standard deviation.

K-means can be considered as a heuristic search algorithm for discovering the cluster assignments that minimize the total sum of squares error. Therefore, there is a chance that a K-means algorithm may converge to the local minimum. To improve the odds of finding the global optimum, we ran 10 iterations with different randomly chosen initializations for each given K value and chose the solution with the lowest sum of squares value.

For deciding on the optimal cluster numbers of the K-means algorithm, we calculated the sum of squared error of each cluster given different K values. However, based on Figure 2.7, there was no significant elbow point that could be used as the optimal K value.

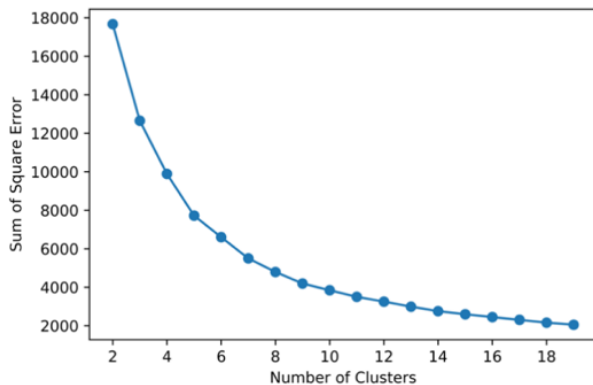


Figure 2.7 The Sum of Square Error for Different K values



Hence, the goodness of fit of the speed model within each cluster was used to determine the optimal cluster number. To establish a robust model across all types of vehicles, we selected the K values which could provide the optimal “lowest adjusted  $R^2$ ” value among all clusters. The relationship between the lowest adjusted  $R^2$  value across all clusters and the cluster numbers is shown in Figure 2.7.

There are two significant elbow points located corresponding to cluster sizes 10 and 18 (Figure 2.8).

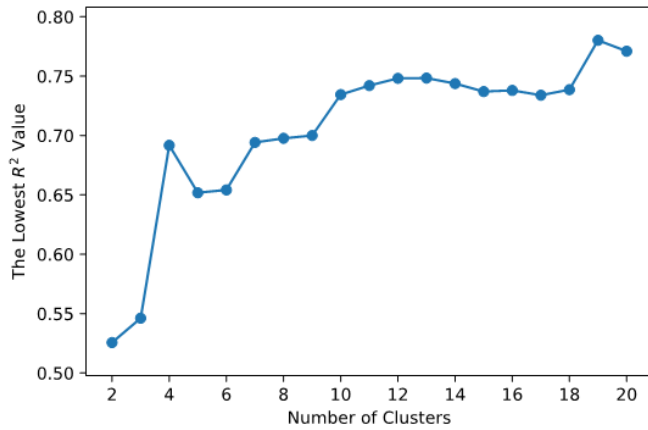


Figure 2.8 The Lowest Adjusted  $R^2$  for Different K values

Although cluster sizes above 18 provided improved adjusted  $R^2$  values, the performance of the multi-layer perceptron models (introduced in the following section) for classifying the single loop signatures degraded significantly above 18 classes (as shown in Figure 2.8). Therefore, the optimal cluster number range was further analyzed from 10 to 18.

A subsequent step was made to further improve the speed estimation model by modifying the feature-weighting K-means algorithm as proposed by Modha and Spangler (Modha & Spangler, 2003). They applied different weights on all the features and set the ratio

of the average within-cluster dispersion and the average between-cluster dispersion along all feature spaces as the objective function. An exhaustive search was performed to determine the best weight combinations which could minimize the objective function (Modha & Spangler, 2003). In this research, the feature-weighting method was adopted to maximize the lowest adjusted R<sup>2</sup> value across all clusters. The feature weightings were:

$$\alpha_1 + \alpha_2 = 1, \alpha_1, \alpha_2 \geq 0 \quad (2.6)$$

Here,  $\alpha_1$  is the weighting on *Lsig*.  $\alpha_2$  is the weighting on *NOMS*.

An exhaustive search was performed to determine the optimal feature weightings on five different settings of cluster numbers, with results shown in Figure 2.9.

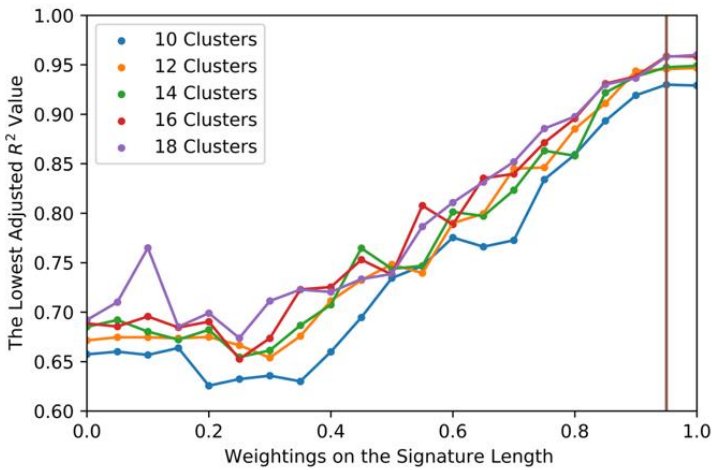


Figure 2.9 Feature Weighting on K-means Algorithm

As the weighting on *Lsig* increased, the lowest adjusted R<sup>2</sup> values across all clusters followed an increasing trend. When the weighting value was above 0.95, the lowest adjusted R<sup>2</sup> values showed a slight reduction. Also, based on the results shown in Figure 2.9, increasing the clusters from 16 to 18 did not present a significant improvement on the performance of the

speed linear regression model. Therefore, the optimal cluster number was determined to be 16 and the optimal weighting was 0.95. A comparison of uniform feature weighting versus optimal feature weighting effects on K-means clustering is presented in Figure 2.10.

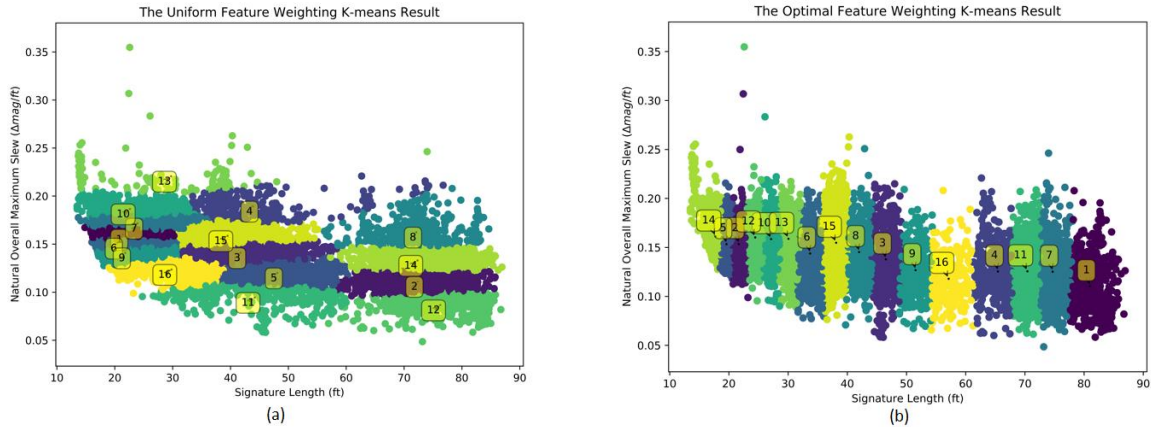


Figure 2.10 K-means Clustering Results

Figure 2.10 shows that the modified weighted K-means algorithm clusters the data predominantly across signature lengths. However, according to the statistical summary of the speed model shown in Table 2.1, the *OMS* is statistically significant for the linear regression model.

Table 2.1 Statistical Summary of Each Linear Model

Cluster	Constant	OMS	$Dur^{-1}$	$R^2$	Cluster Size
14	2.055	-0.473***	18.687***	0.959	2,932
5	1.393	0.728***	17.110***	0.986	4,580
2	2.852	0.845***	18.150***	0.981	2,079
12	2.288	0.806***	20.645***	0.981	2,349
10	1.571	0.479***	24.570***	0.985	1575
13	2.919	0.170***	28.191***	0.980	904
6	2.279	0.177***	31.905***	0.977	706
15	1.366	0.305***	35.666***	0.981	983
8	1.662	0.314***	39.232***	0.981	940
3	0.391	0.382***	43.852***	0.984	667
9	1.347	0.373***	48.268***	0.981	441
16	3.422	0.072***	54.234***	0.971	238
4	2.206	-0.001***	63.680***	0.979	524
11	1.236	0.153***	68.020***	0.986	780
7	1.05	-0.036***	74.075***	0.984	903
1	1.657	-0.008***	79.482***	0.965	613

Note: \*\*\* Significant at p-value < 0.001.

Therefore, the *NOMS* feature was retained in the clustering algorithm.

#### 2.2.4.5 Multiple Linear Regression Model for Each Cluster

The general representation of the linear model with each cluster is shown below.

$$v_{est}^k = \beta_0^k + \beta_1^k \cdot OMS^k + \beta_2^k \cdot Dur^{-1k} \quad (k = 1,2,3 \dots 16) \quad (2.7)$$

Here,  $k$  is the cluster label. We fit a multiple linear regression model over each pre-determined group. The results are shown with the clusters ordered according to their arrangement shown in Figure 2.10b.

Based on the adjusted  $R^2$  values, we found that all the linear models could explain the data well in each cluster. T-tests were adopted to assess the coefficients of each model. The p-values for both terms of all models were lower than 0.05. This indicates that the null hypothesis at  $\alpha = 0.05$  can be rejected, and all the parameters are significant to the linear model.

#### **2.2.4.5 Cluster Classification using Multi-Layer Perceptron (MLP) Model**

Lastly, a supervised learning model was developed to assign the single loop signatures to each pre-determined cluster. A multi-layer perceptron neural network model was designed with 60 features from a single loop signature, with each feature used as the model inputs and the sixteen pre-determined clusters developed from the dual loop sensor data as the model targets.

#### **Feature Extraction for the Multi-Layer Perceptron Model**

Compared to signatures from passenger vehicles, truck signatures possess significantly more diversity. 30 horizontally evenly-spaced magnitude features were extracted from the magnitude-normalized signatures, together with 30 first-order differences obtained from the 30 magnitude features, as the inputs to the classification model. The extracted features are shown in Figure 2.11.

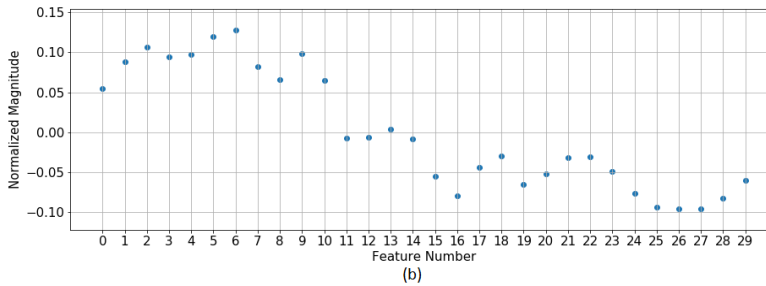
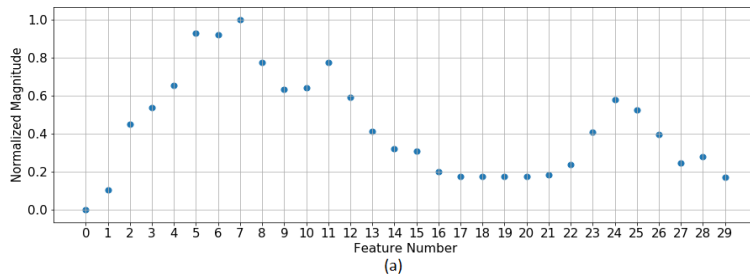


Figure 2.11 Feature Extraction for the Multi-layer Perceptron Model (a) Thirty Magnitude Features (b) Thirty Magnitude Differences Features

### Multi-Layer Perceptron Model

The MLP model is an artificial neural network model with two or more hidden layers. In this research, a feed-forward multi-layer perceptron network was constructed and trained with backpropagation. The MLP model was developed with seven hidden layers and 250 neurons for each hidden layer. The rectified linear unit (*ReLU*) developed by Nair and Hinton (Nair & Hinton, 2010) was used as the activation function.

The overall dataset was separated into two independent datasets: a training and a test dataset. The multi-layer perceptron neural network model was trained using the training dataset with 11,880 observations and evaluated with the test dataset comprised of 6,364 observations. The overall classification accuracy of the training set was 76 percent, with 74 percent for the test dataset.

Based on the model results shown in Table 2.2, the classification performance of the MLP model deteriorated with larger cluster sizes.

Table 2.2 MLP Performance for Different Clustering Method

Method	Optimal feature weighting		Uniform feature weighting	
	Training	Testing	Training	Testing
20	68%	67%	65%	60%
18	69%	67%	60%	56%
16	<b>76%</b>	<b>74%</b>	66%	60%
14	84%	78%	65%	62%
12	83%	79%	74%	69%
10	87%	85%	72%	69%

The results also show that the MLP model performed significantly better with the optimal feature-weighting method.

### 2.1.5 Model Results

The mean absolute error (MAE) and mean absolute percentage error (MAPE) were used as the evaluation metric to assess the performance of the ATISE model. MAE indicates how close the estimated values are to the true speed values. MAPE is used to measure the percentage that estimated speeds deviated from true speeds. Leveraging the truck body type classification model (Hernandez, 2014), we present the performance of the ATISE model by truck body types on the training and test set. We also compared our model with a recent inductive signature-based individual vehicle speed estimation (ISE) model (Tok et al., 2009). The comparisons of these results are shown in Table 2.3.

Table 2.3 Evaluation of Individual Vehicle Speed Estimation on Training and Test Set

	<b>Models</b>	<b>ISE Model (Test)</b>		<b>ATISE Model (Train)</b>		<b>ATISE Model (Test)</b>	
	<b>Measures</b>	<b>MAE (mph)</b>	<b>MAPE (%)</b>	<b>MAE (mph)</b>	<b>MAPE (%)</b>	<b>MAE (mph)</b>	<b>MAPE (%)</b>
100%	Overall	4.18	7.61%	2.33	3.95%	2.45	4.12%
20%	PC	2.56	4.31%	2.03	3.20%	2.08	3.24%
80%	SU	4.39	7.72%	2.48	4.17%	2.54	4.21%
	Single	5.56	10.68%	2.71	4.95%	2.92	5.20%
	Semi	4.66	9.06%	2.00	3.57%	2.32	4.27%
	Multi	5.30	14.00%	1.75	3.78%	2.12	4.31%

Note: PC: Passenger Vehicles, SU: Single Unit Trucks, Semi: Tractors pulling Semi-Trailer, Single: Trucks pulling Single Trailer, Multi: Tractors pulling Multiple Trailers.

For the truck-focused dataset, the performance of the ATISE model is better than the ISE model (Tok et al., 2009) across all vehicle types. The ISE model can estimate passenger vehicle speeds accurately. However, both MAE and MAPE show a significant drop when the model is applied for estimating truck speeds. Conversely, ATISE performs consistently across all vehicle types, even for long trucks such as tractors with multiple trailers. This demonstrates that the ATISE performance is not influenced by the variety of vehicle lengths.

Comparing the test results shown in Figure 2.12 and Figure 2.13, we find that the ISE model tends to overestimate vehicle speeds when the vehicles travel around the free-flow speed.



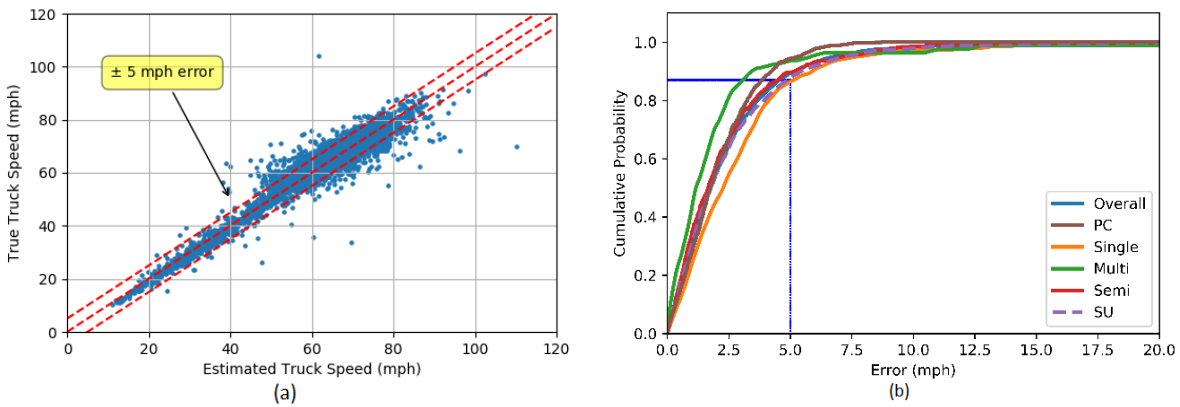


Figure 2.12 Testing Results for the ATISE Model

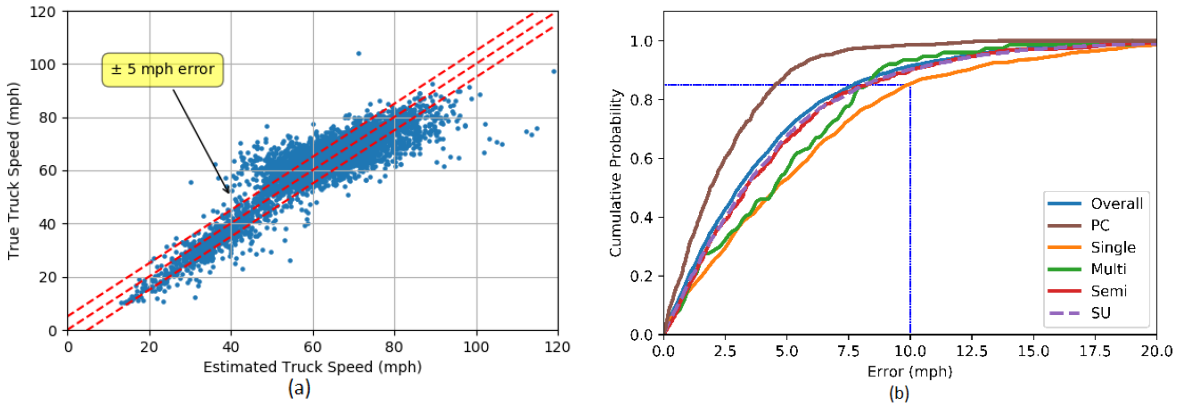
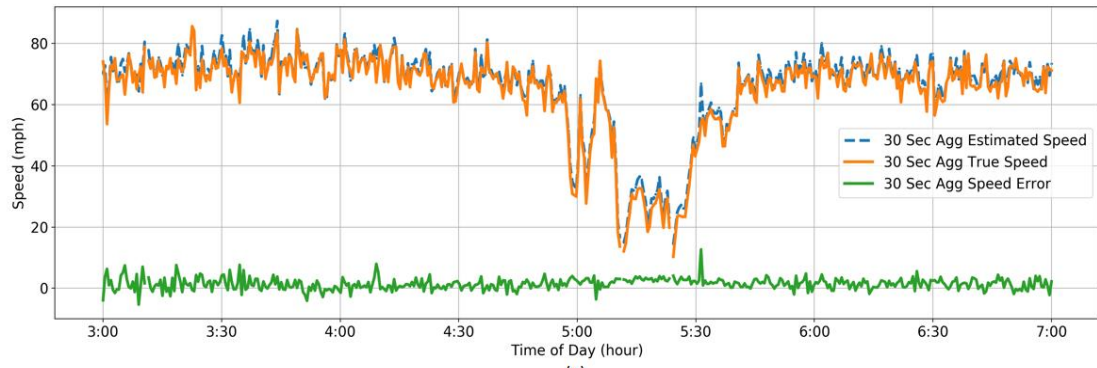


Figure 2.13 Testing Results for the ISE Model

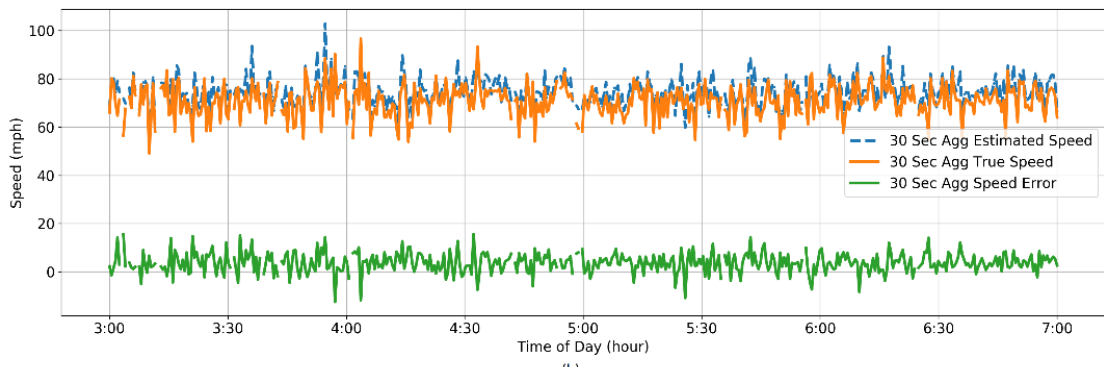
Also, it underestimated vehicle speeds when the vehicles travel at relatively high speeds. Conversely, ATISE performed consistently across all traffic conditions and accurately estimated vehicle speeds within 5 mph error. We compared the cumulative probability distribution between the ISE and ATISE models. Previously, across all vehicle types, at least 80 percent of vehicles had an estimation error of less than 10 mph. In the ATISE model, above 80 percent of vehicles have estimation error less than 5 mph. Again, the ISE model only had a good performance on the passenger vehicle speed estimation. As before, the ATISE model performed consistently across all vehicle types.

In summary, the results confirmed that the ATISE model can provide accurate individual vehicle estimation results across all vehicle types and different traffic conditions.

In this study, a transferability test has been conducted on both the ATISE model and the ISE model. One day of data for transferability testing was collected from the I-580 freeway near the city of Tracy, California. For the ATISE method, the overall individual speed estimation MAE was 3.7 mph and the MAPE value was 5.4 percent. However, the ISE model yielded a higher 9.8 mph MAE and the MAPE value was 13.9 percent. Meanwhile, we tested ATISE and ISE estimation results on 30-second aggregated data with 23 percent truck volumes on a weekday and 16 percent truck volumes on a weekend (Figure 2.14 and Figure 2.15).



(a)



(b)

Figure 2.14 30 Second Aggregated Speed Estimation Result using ATISE Method: (a) 2018-7-13 (Weekday), (b) 2018-7-14 (Weekend)



Figure 2.15 30 Second Aggregated Speed Estimation Result using ISE Method: (a) 2018-7-13 (Weekday), (b) 2018-7-14 (Weekend)

An analysis on a four-hour period that contained both congested and free-flow conditions with 23 percent truck volumes showed that the model overestimated the aggregated speed (Figure 2.15). However, the ATISE model performed well under high truck volumes across various traffic conditions. The MAE of the 30-second aggregated speed estimation using ATISE was 1.56 mph.

### 2.1.6 Conclusion

This section proposed a truck-focused individual speed estimation (ATISE) model. This new method adopted a similar model structure from previous studies (Sun & Ritchie, 1999; Park & Ritchie, 2010; Tok et al., 2009). However, improvements were made to the signal

processing procedure, feature extraction, and clustering approach. This new speed estimation model has demonstrated promising results on a variety of traffic conditions under high truck volumes, and at an independent test location. Moreover, the ATISE model performed consistently well across different vehicle types.

This new model has been compared to the previous model (ISE)(Tok et al., 2009) and showed a significant improvement in the accuracy of individual truck speed estimation. With accurate individual vehicle speed estimation results, this model was able to accurately estimate the 30-second aggregated speeds on a roadway section, even with high truck volumes.

This section helps fill the gap on accurate individual-level truck speed estimation. Furthermore, this new model can be integrated into the existing California Truck Activity Monitoring System (Tok et al., 2017), which can provide detailed truck classification information at a statewide level. Accurate truck speeds and detailed truck classification can be helpful for emissions estimation (United States Environmental Protection Agency, n.d.), safety evaluation (Malyshkina & Mannering, 2008), and driving behavior analysis (Daganzo, 2002).

## **2.2 A Deep Ensemble Neural Network Approach for FHWA Axle-based Vehicle Classification using Advanced Single Inductive Loops**

### **2.2.1 Introduction**

FHWA's Traffic Monitoring Guide outlines a standardized classification scheme to serve various transportation needs. The FHWA vehicle classification scheme categorizes vehicles into thirteen classes based on tire and axle combination while partially taking into account general body configuration (Federal Highway Administration, 2013). This classification scheme has been widely used for pavement design to account for the dissimilar pavement impacts attributed to physical vehicle characteristics such as axle loads, spacing, and tire configuration (Gillespie, T.D, Karamihas, S.M. & Sayer, 1993). In addition, aggregated FHWA vehicle classes have been used as input for on-road emission estimation models (Guensler et al., 2005) as well as in freight forecast modeling (Beagan et al., 2019; Schaefer, Ron; Worth, Monica; Heilman, Jonathan; Kehoe, 2017)

Since truck size and weight laws vary by states and truck configuration populations vary across states (Federal Highway Administration, 2019), some state agencies modify the FHWA's 13 categories to meet their transportation application needs (Federal Highway Administration, 2013). For example, Class 9 type 32 trucks in California are distinguished from other Class 9 trucks in the FHWA 13-Category Scheme and form a standalone class labeled as Class 14 (Quinley, 2010). Since the data used in this dissertation was collected in California at the statewide level, the model was focused on classifying vehicles into the California-modified FHWA scheme (FHWA-CA). Table 2.4 provides a brief description of each class in the FHWA-CA classification scheme.

Table 2.4 FHWA-CA classification scheme definitions (FHWA (Federal Highway Administration), 2014)

FHWA-CA Class	Vehicle Description	Class Includes	# of axle
1	Motorcycle	Motorcycles	2
2	Passenger Vehicles	All cars, Cars with one-axle trailers, Cars with two-axle trailers	2, 3 or 4
3	Other two-axle four-tire Single-unit Vehicle	Pickups and vans, Pickups and Vans with one- and two-axle trailers	2, 3 or 4
4	Bus	Two- and three-axle buses, Bus with trailer	2 or 3 (tractor)
5	Two-axle, Six-tire, single-unit trucks	Two-axle trucks, two-axle trucks with trailer	2 (tractor)
6	Three-axle single-unit trucks	Three-axle trucks, Three-axle tractors without trailers	3
7	Four or more axle single-unit trucks	Four-, five, six- and seven-axle single-unit trucks	4 or more
8	Four or fewer axle single-trailer trucks	Two-axle trucks pulling one- and two-axle trailers, Two-axle tractors pulling one- and two-axle trailers, Three-axle tractors pulling one-axle trailers	3 or 4
9	Five-axle single-trailer trucks	Two-axle tractors pulling three-axle trailers, Three-axle tractors pulling two-axle trailers, Three-axle trucks pulling two-axle trailers	5
10	Six or more axle single-trailer trucks	Three-axle tractors pulling three-axle trailers	6 or more
11	Five or fewer axle multi-trailer trucks	Multiple configurations (Multi-unit trucks)	4 or 5
12	Six-axle multi-trailer trucks	Multiple configurations (Multi-unit trucks)	6
13	Seven or more axle multi trailer trucks	Multiple configurations (Multi-unit trucks)	7 or more
14	Single and tandem axle on tractor, single and single axle on the trailer	Single and tandem axle on tractor, single and single axle on the trailer	5
15	Unclassified vehicles	Multiple configurations	2 or more

Weigh-In-Motion (WIM) and Automatic Vehicle Classification (AVC) sites using axle sensor technologies can directly capture vehicle axle configuration information. Hence, these types of systems have been commonly used to report vehicle classification counts according to FHWA-based schemes. However, those sensors are not comprehensively deployed in the transportation highway network due to their high installation and maintenance costs. Conversely, inductive loop sensors are much more widely deployed in many jurisdictions as they have a much lower installation and maintenance cost. Studies have investigated the use of inductive vehicle signatures to classify vehicles based on the FHWA classification scheme (Jeng et al., 2013; Jeng & Ritchie, 2008). However, a closer evaluation of these efforts showed

that the performance of these models was skewed towards non-trucks—the high accuracy in classifying passenger vehicle classes and Class 9 trucks conceals the deficiencies of the models in identifying other FHWA truck-related classes. Even though trucks generally account for approximately 5 to 20 percent of traffic streams, the adverse impact of misclassifying trucks could be significant. For instance, implementation of a biased model may underestimate the pavement damage caused by trucks, since pavement structures are disproportionately impacted by heavy trucks (Gillespie, T.D, Karamihas, S.M. & Sayer, 1993). From a planning perspective, unreliable truck counts may result in a flawed understanding of truck activities and misinformed policy decisions to manage future demand for truck movements and operations. Furthermore, the class bias issue also occurs within truck-related classes. Certain types of trucks in the FHWA classification scheme – such as Classes 7, 11, 12 and 13 – are not as frequently on the roadway network. On the other hand, Class 9 trucks are the most common multi-unit truck configuration observed along most corridors and show great variability in their body types. However, the basic assumption of canonical machine learning algorithms for classification problem is that the number of training instances in considered classes are relatively similar (Krawczyk, 2016). Consequently, the imbalanced dataset poses a natural difficulty for many classification algorithms to correctly classify minority classes, since they are naturally biased towards majority classes. Notwithstanding, some of the minority classes remain prominent components in both pavement design (Gillespie, T.D, Karamihas, S.M. & Sayer, 1993) and freight forecast modeling (Beagan et al., 2019; Schaefer, Ron; Worth, Monica; Heilman, Jonathan; Kehoe, 2017).



To address this gap, this section details the development of an accurate and transferable FHWA vehicle classification model using a truck-focused dataset. The model shows significant improvement over previous signature-based FHWA vehicle classification models (Jeng et al., 2013; Jeng & Ritchie, 2008) for all truck classes in terms of F1 scores, especially on minority classes such as Classes 7 and 11 which were overlooked by previous studies (Jeng et al., 2013; Jeng & Ritchie, 2008). The algorithm development comprised three steps. First, the training and testing dataset was split using stratified sampling to retain the representativeness of the training instances for minority classes. Then, a deep neural network (DNN) with dropout layers was constructed to reduce the generalization error. Finally, a bootstrap aggregating ensemble (bagging) was developed to address the classification challenge with an imbalanced dataset. In this step, bootstrap resampling was applied on the training set to approximate the feature distribution of the under-represented classes to facilitate a better understanding of those classes with limited training instances to learn from. The bagging DNN successfully improved the model performance on minority classes without compromising the accuracy of majority classes. The spatial and temporal transferability of the model was empirically tested using independent datasets.

### **2.2.2 Literature Review**

The state-of-the-art methods that were used to obtain FHWA axle-classification data can be summarized into two categories: axle sensor- and single inductive loop-based methods.

#### **Axle Sensor Methods**

FHWA axle-based vehicle classification data have been traditionally collected via axle-based sensors, such as road tube arrays (Beagan et al., 2019), piezoelectric sensors (Bitar & Refai, 2017), WIM systems (Kwigizile et al., 2005), and wireless accelerometer sensors (W. Ma et

al., 2014), all of which have the ability to capture axle numbers and spacing configurations. Department of Transportation agencies across the U.S. rely on existing classification sites equipped with such axle-based sensors for reporting FHWA vehicle classification counts. Efforts have been devoted to testing the performance of classification sites and investigating methods to enhance the classification accuracy on these systems, which yield their own limitations in distinguishing classes with overlapping axle configurations (Bitar & Refai, 2017; Kwigizile et al., 2005). For example Classes 2 through 5 include two-axle vehicles (Kwigizile et al., 2005) and the first axle-spacing of Classes 3, 5, and 8 share overlapping range values (Bitar & Refai, 2017). Kwigizile et al. improved the correct classification rate for overlapping classes by breaking down the 13 FHWA classes into 28 detailed subclasses and using a probabilistic neural network to assign axle spacing values from a calibrated WIM site to those predefined subclasses (Kwigizile et al., 2005). This model reduced the error rate of the calibrated WIM site from 9.5 percent to 6.2 percent. The author highlighted that the misclassification came from systematic errors due to overlapping axle configurations across FHWA classes. By introducing weight value as one of the input variables, the error rate was further reduced to 3.0 percent. However, weight values vary spatially and temporally and may affect the transferability of their model. Bitar et al. adopted a probabilistic approach to improve the accuracy of the classification site equipped with piezoelectric sensors (Bitar & Refai, 2017). In their study, a comprehensive classification error analysis was conducted on the overlapping axle configuration across the FHWA scheme's categories. Their hypothesis was that axle spacing distributions are different for classes that may share similar axle configurations. Hence, the axle spacings associated with each class were fitted into Gaussian distributions. Subsequently, the optimal class boundary thresholds for the overlapping axle

configuration were determined according to the estimated axle-spacing distributions. The error rate was significantly reduced from the original sensor outputs especially for Classes 3, 6, and 7 (Bitar & Refai, 2017). However, axle-based sensors do not provide any truck general body type related information defined by the FHWA classification scheme. Therefore, it is a challenge for axle detectors to accurately differentiate trucks with overlapping axle-spacing distribution, even though they have distinctly different body types. For example, the error rate of their model on Class 4 (bus or bus with a trailer) was higher compared to the original sensor measurements. The first-spacing probability density distribution of Classes 4 and 5 had a significant overlap and did not present a clear class decision boundary, although those distributions came from two distinct body types -- buses in Class 4 and single-unit trucks in Class 5.

In addition, a prototype wireless accelerometer system, which detects the pavement vibration when vehicles traverse the detection area, has also been explored for estimating axle-based classification (W. Ma et al., 2014). The sensor identified the axle configuration by locating the vibration peaks. The accelerometer-based classification was evaluated using calibrated WIM classification results. This wireless sensor was able to achieve the same level of accuracy as the current WIM system on estimating FHWA classes. However, accelerometer sensors would experience the same limitations as other axle detectors in distinguishing classes with overlapping axle spacing ranges.

The detection sites equipped with WIM systems or piezoelectric sensors are sparsely deployed in the roadway network due to their high installation cost. Hence, researchers

investigated existing inductive loop sensors as an alternative, since they are widely deployed across the U.S and their installation and operation costs are relatively cheap.

### Single Inductive Signature Methods

Unlike conventional inductive loop detector cards which sample at a rate of 240 Hz (Coifman & Neelisetty, 2014), advanced loop detectors capture detailed inductance changes when a vehicle traverses an inductive loop sensor. The resulting high-resolution waveform generated by the inductance change measurements is known as an “inductive vehicle signature” (Sun & Ritchie, 1999). However, the exact axle locations cannot be directly identified from inductive vehicle signatures (Figure 2.16), which is one challenge in classifying vehicles into the FHWA scheme.

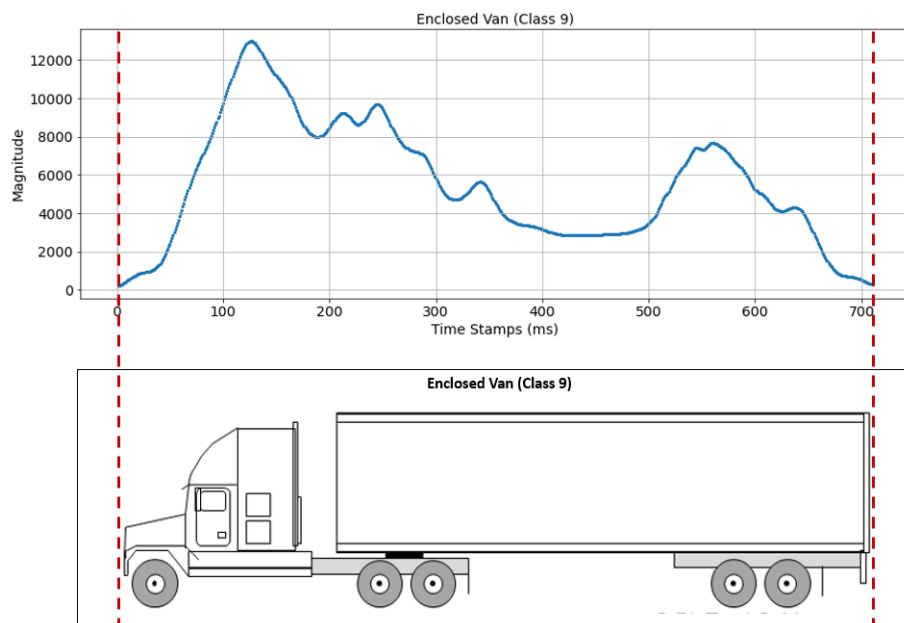


Figure 2.16 Class 9 Enclosed Van and its corresponding raw signature

Jeng and Ritchie made the first attempt to classify vehicles on the basis of the FHWA scheme using single inductive loop signature data (Jeng & Ritchie, 2008). The piecewise

slope rates (PSR) of each interpolated signature were used as a reduced representation of each signature pattern. The PSRs of each signature were separated into five groups by visual observation of PSR plots of all vehicle classes (Jeng, 2007). Unfortunately, due to the data limitation, Class 10 to Class 13 trucks, which have disproportionately severe negative impacts on pavement structure, were not considered in their model development process. Later, Jeng et al. enriched their dataset with multi-unit trucks and proposed a new vehicle classification algorithm (Jeng et al., 2013). The new algorithm is composed of two steps. First, vehicle signatures were transformed and reconstructed with wavelet transformation. Then, the transformed vehicle signatures were grouped into FHWA classes using K nearest neighbor. Even though the overall accuracy was 92 percent, the weakness on classifying minority classes was obscured by the high performance on predicting the majority classes (Class 2 and Class 3), which overlooked the poorer performance on several truck classes (Class 6, 7, 8, 11, 12, 13). Hence, the focus of this section is to address the effect of dataset imbalance on the performance of minority classes.

### **2.2.3 Data Description**

The vehicle signature data used in this paper were collected at 20 different detection sites equipped with either 6ft square or 6ft round inductive loop sensors across California in 2012, 2013, and 2016. The geographical distribution of the 20 data collection sites is shown in Figure 2.17.



Figure 2.17 Data Collection site for model training, hyperparameter tuning, and transferability testing

The selected detector sites experienced high truck volumes, a wide variety of truck types, and the data collection effort spanning various traffic conditions. A total of 44,438 vehicle signature records were processed primarily at the truck lanes in each facility, with a resulting vehicle class distribution as shown in Table 2.5.

Table 2.5 Vehicle Class Distribution from ground truth

<b>FHWA-CA Scheme</b>	<b>Counts</b>	<b>Imbalance Rate</b>	<b>Number of Body Types</b>
1	2	0.0001	1
2	2,946	0.1494	4
3	8,203	0.4159	2
4	772	0.0391	3
5	7,055	0.3577	32
6	1,535	0.0778	27
7	279	0.0142	9
8	1,463	0.0742	24
9	19,724	1.0000	40
10	138	0.0070	16
11	1,518	0.0770	21
12	268	0.0136	12
13	3	0.0002	1
14	764	0.0387	14

The imbalance rate presented in Table 2.5 was the metric used to understand the quantitative relationship between majority and minority classes. It is defined as the ratio of each minority class to the majority class (Class 9). One key challenge in the model development was to accommodate the imbalanced dataset. When training an imbalanced dataset, models are generally prone to enhance the prediction accuracy of the majority class (Krawczyk, 2016). This will lead the majority classes and the overall model to achieve relatively high prediction accuracy at the expense of minority classes (Krawczyk, 2016).

Typically, the issue of imbalanced datasets in classification is addressed at either the data or algorithm level (Aouatef Mahanil; Ahmed Riad Baba Ali, 2016). At the data level, minority classes are typically oversampled, while majority classes are undersampled to

balance the dataset. However, undersampling may compromise the generality of the model. For instance, Class 9 is a majority class comprising heterogeneous body types with distinct signature waveforms, as shown in Figure 2.18.

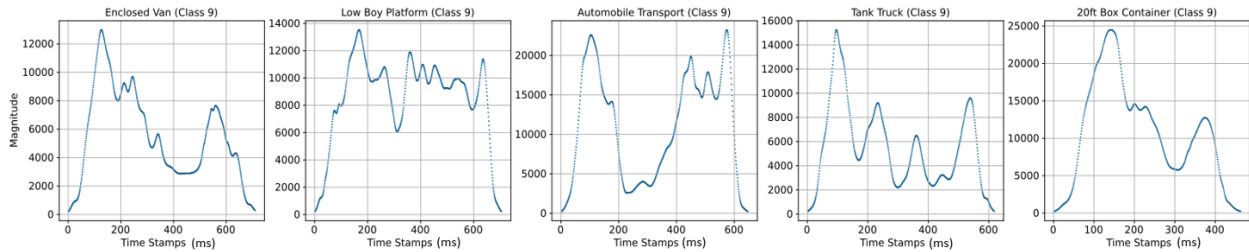


Figure 2.18 Signatures of different truck body types

Undersampling may cause the information to be lost by removing unique vehicle configurations that would have helped the model to better capture the characteristics of diverse vehicles found in this class. On the other hand, synthetic data methods which are generally used in oversampling could create overlapping instances between the minority class and the majority class, and further reduce the prediction accuracy for the majority class (Aouatef Mahanil; Ahmed Riad Baba Ali, 2016). Therefore, this study investigated algorithm-level enhancements to improve the performance of the signature-based FHWA classification model.

According to Table 2.5, both Classes 1 and 13 have training instances extremely small and less than thirty, which are empirically considered as insignificant sample sizes. Moreover, this research primarily focused on the FHWA-CA truck-related classes. Therefore, Classes 1 (motorcycle) and 15 (unclassified vehicle) were excluded in the modeling process. Since Class 13 shares similar body types as well as axle configurations with Class 12 vehicles,



they were combined in the model. Classes 1 and 15 (Unclassified vehicle) may be included and Classes 12 and 13 may be split in the future with further enrichment of the dataset.

## **2.2.4 Model Development**

Prior to model development, stratified sampling was initially used to partition the training and testing datasets with a 70-30 split, respectively, in order to ensure that a sufficient number of samples for each class can be observed in both the training and testing sets.

### **2.2.4.1 Feature Extraction**

First, raw signatures were processed using cubic spline interpolation to eliminate noises and obtain a set of feature vectors with the same dimension and normalized on the vertical (magnitude) axis (Jeng, 2007). This yielded a vector of 31 magnitude features equally spaced along the normalized time domain with 30 degrees of freedom. Subsequently, 30 differences were derived from 31 magnitude values and then further normalized along the y-axis, which forced magnitude and difference values to fall into the same scale. Finally, the last value of the vector of 31 magnitudes was dropped to retain the independence of the feature vector. This resulted in 60 (30 magnitudes and 30 differences) independent features that were used as inputs for the vehicle classification model. The feature extraction process is presented in Figure 2.19.

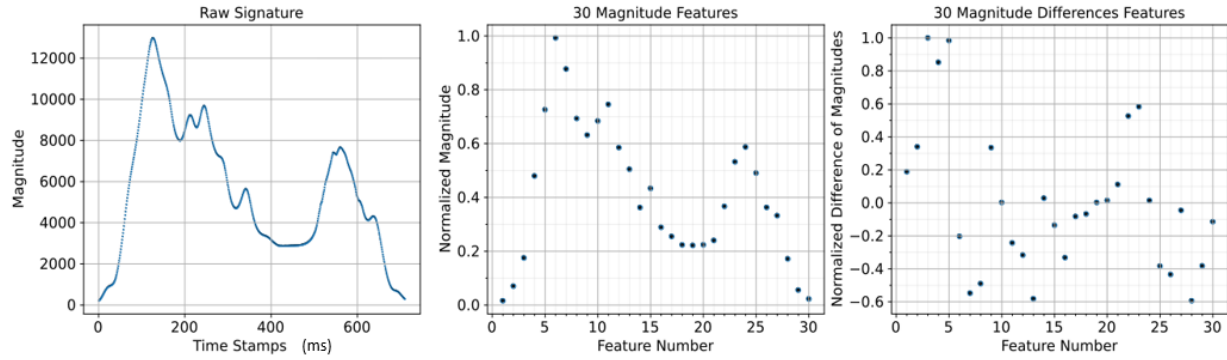
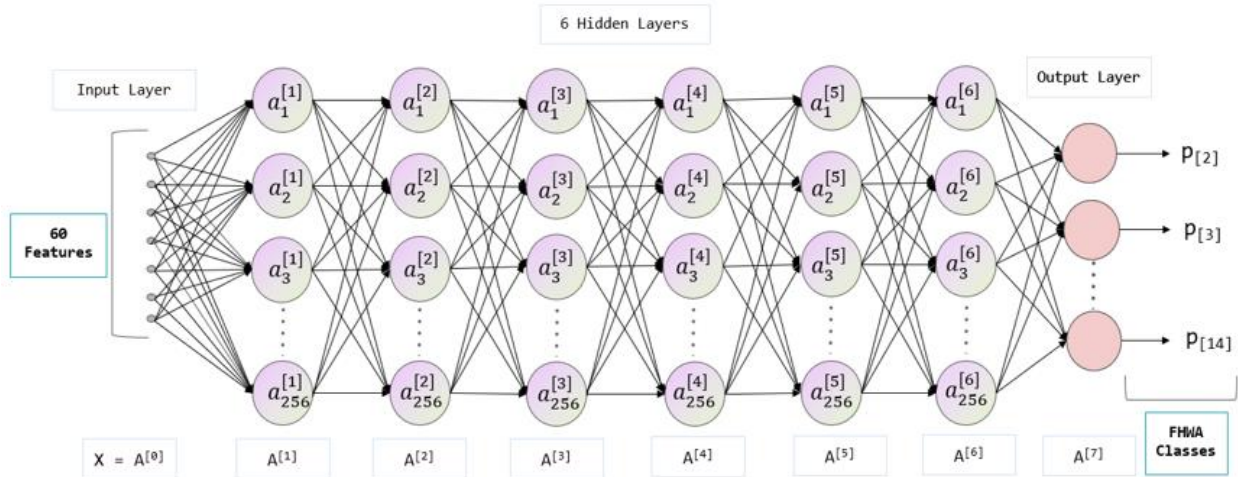


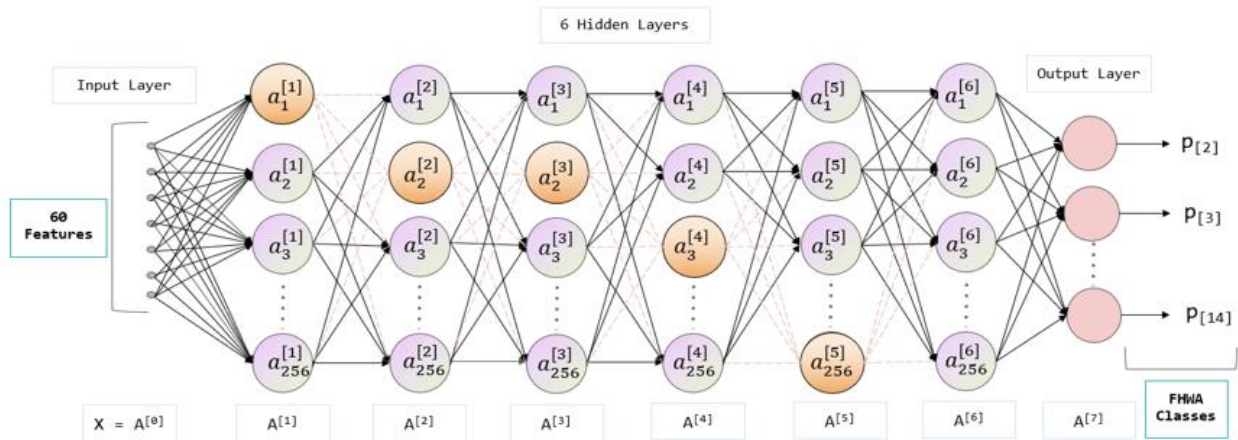
Figure 2.19 Preprocessing and Feature Extraction

### 2.2.4.2 Deep Neural Network Architecture

Theoretically, the multiple layer structure with non-linear activation of a deep neural network allows it to approximate the shape of complex mapping functions (Goodfellow et al., 2016), which is suitable for the task of multi-class classification problems. Therefore, this study adopted a deep neural network model with dropout regularization to classify vehicles based on the FHWA-CA scheme. The model was constructed with 6 hidden layers, 256 neurons on each hidden layer (shown in Figure 2.20). The Rectified Linear Unit (ReLU) (Nair & Hinton, 2010) was used as the activation function on each hidden layer, while the Softmax activation function was applied on the output layer to represent the probability distribution over the 12 FHWA categories (where Classes 12 and 13 were combined). To avoid gradients vanishing and exploding, He (He et al., 2015) and Xavier (Glorot & Bengio, 2010) weight initialization methods were applied to the hidden layers with ReLU and Softmax activation functions, respectively. The deep neural network model was trained with a minibatch size of 100 and a learning rate of 0.001. The Adam optimizer (Kingma & Ba, 2015) was adopted to solve this highly non-linear optimization problem.



(a) The Deep Neural Network Architecture



(b) The Deep Neural Network with Dropout Layers

Figure 2.20 Model Structure

Balancing bias and variance of the deep neural network model is an essential task during the hyperparameter tuning process. Bias represents the expected deviation from the true value of the function while variance measures the deviation from the expected estimator which is caused by the unseen dataset. The deep neural network model is typically prone to overfit the training set as models increase in complexity, especially for the majority classes. This results in a trained model with low bias and high variance as shown in Figure 2.21a, where the training error tends to decrease (low bias) and the testing error tends to increase (high

variance). Dropout regularization—a computationally inexpensive but powerful regularization method for deep neural network models—was implemented at each layer in the deep neural network to prevent overfitting (Mele & Altarelli, 2014). Thirty percent of the neurons within each hidden layer were randomly dropped out while the remaining 70 percent were retained (Figure 2.20b) during the training process. A comparison of early stopping without and with dropout regularization is shown in Figure 2.21c and Figure 2.21d, respectively. The training process needed to be terminated significantly earlier at 5 epochs without dropout regularization. This resulted in a poorer test data prediction accuracy of 0.87, compared with 0.91 for the latter.

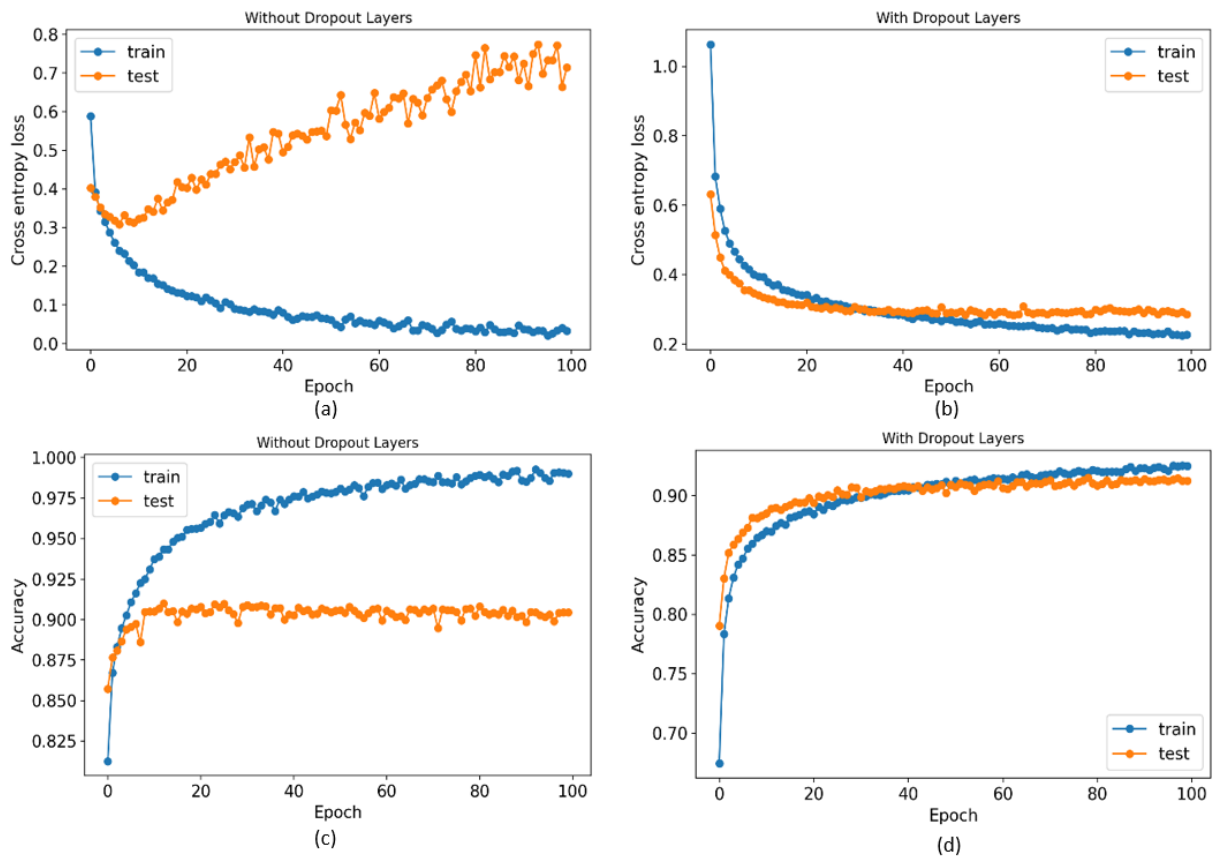


Figure 2.21 Learning Curve

### 2.2.4.3 Bootstrap Aggregating

As Table 2.4 shows, the labeled FHWA classes yielded an imbalanced class distribution. The number of instances belonging to a certain class, such as Class 9, was significantly higher than any other labeled class in the dataset. The objective function of the designed neural network model is to minimize the global error rate. The entire cost function for a multi-class classification problem given  $m$  training instances labeled with  $n$  classes can be written as:

$$J = \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m} \sum_{i=1}^m (y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})) \quad (2.7)$$

where,  $y^{(i)}$  and  $\hat{y}^{(i)}$  represents the true class and the predicted class of training instances  $i$  respectively.

As Equation 2.7 shows, the cost function does not handle the class distribution in the dataset. Without enough training instances to approximate the feature distributions of minority classes, the model is inclined to compromise the performance of minority to achieve a low global error rate resulting in the poor performance of minority classes.

In order to have a better understanding of feature distributions within each class, a bootstrap aggregating ensemble was adopted for the model development. Bootstrap aggregation (Bagging) is an ensemble strategy used to enhance the generalizability of the model through the combination of several models trained by multiple bootstrap samples (Breiman, 1994). The basic idea of Bagging comes from the bootstrapping resampling technique, which is used to approximate the empirical distribution of the observed data, especially for datasets with small class samples. Stratified bootstrapping was applied on the training set and ten sets of bootstrapped samples were formed by resampling the stratified training instances with

replacement. The bootstrap samples were fed into the DNN model with the same model structure. Subsequently, the prediction scores from ten models were averaged and the class corresponding to the highest averaged prediction score was considered as the final decision.

The bagging DNN model structure is shown in Figure 2.22.

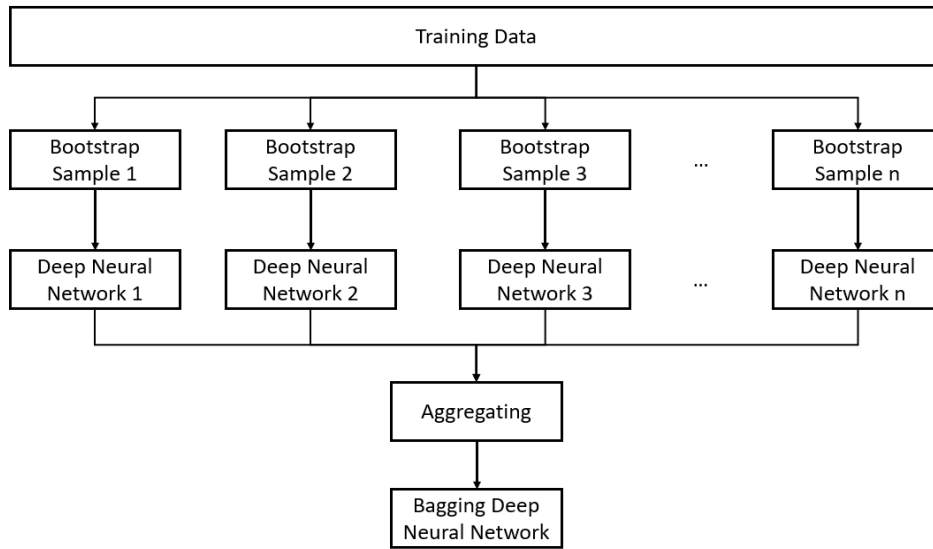


Figure 2.22 Illustration of Bagging DNN

## 2.2.5 Model Results

### 2.2.5.1 Evaluation Metrics

Determining appropriate performance measures is an essential task for evaluating the model built upon an imbalanced dataset. The accuracy measure (Equation 2.8)—which is derived from a confusion matrix (Table 2.6)—presents the percentage of total instances being correctly classified.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.8)$$

Table 2.6 Generic Confusion Matrix

		Predicted Class	
		Positives	Negatives
Actual Class	Positives*	True Positives (TP)	False Negatives (FN)
	Negatives*	False Positives (FP)	True Negatives (TN)

Note: For the illustration purpose, the positive cases were assumed to be the minority class and negative cases were the majority class

The accuracy measure is determined by both  $TP$  and  $TN$  (Table 2.6) on the numerator of the fraction. If the  $TN$ , which represents the correctly classified instance from the majority class, is disproportionately larger than  $TP$ , the accuracy value will still be large, even though the performance on the minority class remains poor. Therefore, this accuracy measurement is sensitive to class skews and is not a reasonable metric for selecting models developed on an imbalanced dataset (Joshi, 2002).

This section discusses three metrics that have generally been used to evaluate the performance of classification models built with imbalanced datasets. Precision (Equation 2.9) and recall (Equation 2.10) are two basic metrics, which are directly calculated from the confusion matrix (Table 2.6). Both precision and recall do not involve the true negative value, which represents the number of majority instances being correctly classified. Hence, these two metrics evaluate the model performance of majority and minority classes independently. The F1 score in Equation 2.11 is the harmonic mean of precision and recall. This metric accounts for precision and recall simultaneously while evaluating the models. In order to evaluate the models developed using imbalanced datasets, the F1 score was primarily used in this paper for the model comparison.

$$Precision = \frac{TP}{TP + FP} \quad (2.9)$$

$$Recall = \frac{TP}{TP + FN} \quad (2.10)$$

$$F1\ score = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (2.11)$$

### 2.2.5.2 Results Analysis

Figure 2.23 shows the recall distribution – which is also referred to as “Correct Classification Rate” (CCR) in previous research (Hernandez et al., 2016; Sahin et al., 2020) – for the DNN models built with 10 sets of bootstrapped samples. Minority classes such as Classes 4, 7, 10, and 12 & 13 resulted in relatively high prediction variances since limited training instances were used to learn from the features for every single model. Therefore, the bagging ensemble model was needed.

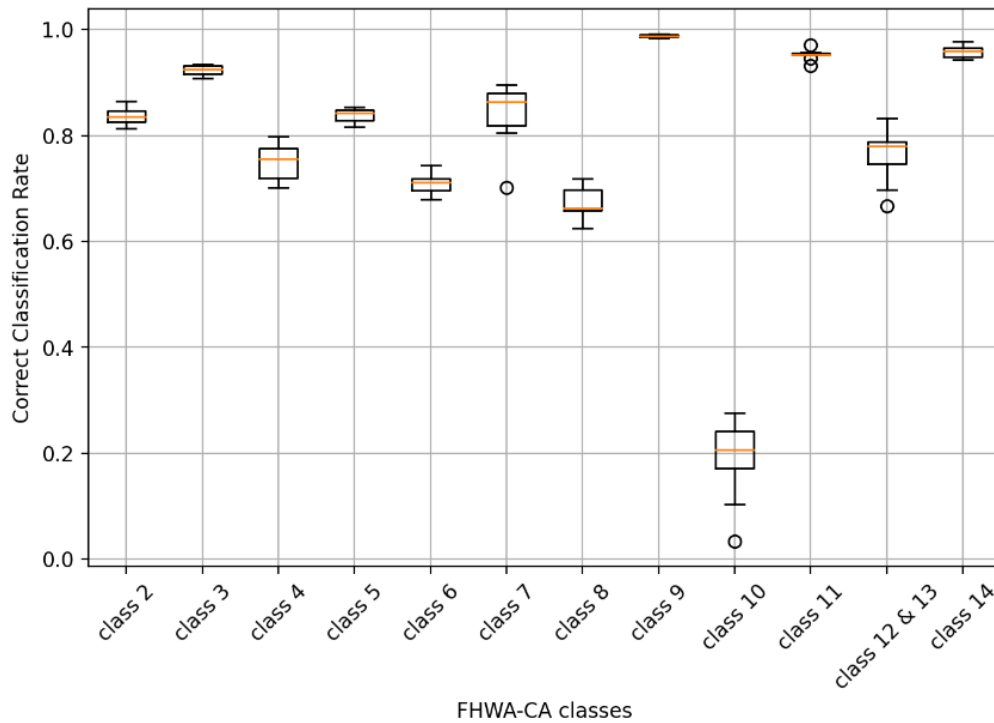


Figure 2.23 Correct Classification Rate across all Classes



Table 2.7 shows the F1 score for each class of three model structures. The dropout technique enhanced the generality of the DNN model and improved the F1 score on the testing set for all classes. With bagging ensemble, the model performance improved across most of the minority classes without compromising the model performance on the majority classes. The overall accuracy of the final model was 0.92 and the average F1 score was 0.83. This bagging DNN was applied on a spatially and temporally independent detection site and was still able to achieve an accuracy of 0.87 and an average F1 score of 0.72.

Table 2.7 Test Result Comparison

	<b>F1 Score of Single DNN Without Dropout</b>	<b>F1 Score of Single DNN With Dropout</b>	<b>F1 Score of Bagging DNN</b>	<b>Test Samples</b>
<b>Class 2</b>	0.85	0.85	0.87	847
<b>Class 3</b>	0.87	0.88	0.89	2,057
<b>Class 4</b>	0.71	0.81	0.83	268
<b>Class 5</b>	0.82	0.85	0.87	1,579
<b>Class 6</b>	0.72	0.80	0.80	362
<b>Class 7</b>	0.77	0.86	0.84	77
<b>Class 8</b>	0.69	0.73	0.78	427
<b>Class 9</b>	0.98	0.98	0.99	4,318
<b>Class 10</b>	0.22	0.35	0.36	29
<b>Class 11</b>	0.96	0.94	0.96	276
<b>Class 12 &amp; 13</b>	0.78	0.79	0.81	66
<b>Class 14</b>	0.96	0.97	0.97	175
<b>Accuracy</b>	0.89	0.91	0.92	10,481
<b>Average F1 score</b>	0.78	0.82	0.83	10,481

The performance of the bagging DNN approach was also compared with a state-of-the-art Wavelet-KNN -based classification algorithm developed by Jeng et al.(Jeng et al., 2013). Jeng et al. evaluated their model performance on each class using CCR. For the overall model performance, the accuracy value was selected as the evaluation metrics in their paper. However, such accuracy is a bias towards the majority class, which was Class 2 in their dataset. Therefore, the F1 scores of their model were recalculated for a fair comparison. As

shown, the bagging DNN model achieves the same level of accuracy in terms of the accuracy metric. Considering the F1 score, the bagging DNN outperform the previous model (Jeng et al., 2013). Except for Class 2, the F1 score for all classes are significantly higher than the previous approach (Jeng et al., 2013). This indicates that the imbalanced dataset issue was well-managed by the bagging DNN model.

Table 2.8 Model Comparison

	Bagging Deep Neural Network			Wavelet-KNN (Jeng et al., 2013)		
	Recall (CCR)	F1 Score	Testing Samples	Recall (CCR)	F1 Score	Testing Samples
<b>Class 1</b>	N/A	N/A	N/A	0.83	0.81	74
<b>Class 2</b>	0.85	0.87	847	0.97	0.97	11,177
<b>Class 3</b>	0.93	0.89	2,057	0.78	0.79	1,568
<b>Class 4</b>	0.78	0.83	268	0.59	0.10	17
<b>Class 5</b>	0.86	0.87	1,579	0.67	0.72	543
<b>Class 6</b>	0.75	0.80	362	0.48	0.39	65
<b>Class 7</b>	0.87	0.84	77	0.67	0.13	3
<b>Class 8</b>	0.71	0.78	427	0.46	0.33	48
<b>Class 9<sup>1</sup></b>	0.99	0.99	4,318	0.86	0.91	754
<b>Class 10</b>	0.17	0.36	29	0.67	0.11	3
<b>Class 11</b>	0.97	0.96	276	0.58	0.26	14
<b>Class 12 &amp; 13<sup>2</sup></b>	0.76	0.81	66	0.75	0.50	4
<b>Class 14</b>	0.97	0.97	175	N/A	N/A	N/A
<b>Accuracy</b>	0.92		10,481	0.92		14,270
<b>F1 score</b>	0.83		10,481	0.50		14,270
<b>Weighted Average F1 Score (1:2)</b>	0.82		10,481	0.47		14,270

Note: <sup>1</sup> Class 9 in the Wavelet-KNN model was combined with Class 14. <sup>2</sup> Classes 12 and 13 were split in the Wavelet-KNN model with CCR of 1.00 and 0.50 respectively, and with an F1 score of 0.45 and 0.50 respectively.

Since trucks have a disproportionately negative impact on pavement structures (Gillespie & Karamihas, 2009), having an accurate prediction result on truck-related classes (from Class 5 to Class 14) is essential for effective pavement design. Therefore, these two models were also evaluated using weight average, where truck-related classes were assumed to be at least two times more important than passenger vehicles. As Table 2.8

shows, the bagging DNN model was superior to the state-of-the-art signature-based FHWA vehicle classification algorithm on predicting truck-related class. The bagging ensemble model achieved a weighted average F1 score of 0.82, where the Wavelet-KNN model only had a value of 0.47.

### 2.2.5.3 Error Analysis

As Table 2.9 shown below, the model presented in this study achieved an F1 score greater than 0.80 for most classes, with the exception of Classes 8 and 10. According to the confusion matrix in Table 2.9, 8.7 percent of Class 8 vehicles are misclassified as Class 3 and 4.7 percent of Class 8 vehicles are misclassified as Class 5. This is mainly caused by the overlapping body types across Classes 3, 5, and 8.

Table 2.9 Confusion Matrix for Test Set

	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10	Class 11	Class 12&13	Class 14	Testing Samples	F1 Score
Class 2	716	122	0	8	0	0	1	0	0	0	0	0	847	0.87
Class 3	75	1,921	1	44	0	0	13	3	0	0	0	0	2,057	0.89
Class 4	1	6	208	39	7	1	4	2	0	0	0	0	268	0.83
Class 5	6	165	12	1,359	23	1	10	3	0	0	0	0	1,579	0.87
Class 6	1	0	8	72	271	9	1	0	0	0	0	0	362	0.80
Class 7	0	0	0	3	7	67	0	0	0	0	0	0	77	0.84
Class 8	0	37	4	20	0	0	303	63	0	0	0	0	427	0.78
Class 9	0	1	0	8	0	0	18	4,286	3	0	0	2	4,318	0.99
Class 10	0	0	0	0	0	0	3	20	5	0	0	1	29	0.36
Class 11	0	0	0	1	1	0	0	1	0	267	6	0	276	0.96
Class 12 & 13	0	0	0	0	0	0	1	1	0	13	50	1	66	0.81
Class 14	0	0	0	0	0	0	0	4	1	1	0	169	175	0.97

Note: Yellow cells indicate correct classifications by class. Grey cells highlight significant misclassifications.

These three classes are also hard to be distinguished using current classification sites (FHWA (Federal Highway Administration), 2014). As Figure 2.24 and Figure 2.25 present, some Classes 3, 5 and 8 trucks share very similar body types and axle configurations on their drive unit. Therefore, it remains a challenge to classify these vehicles accurately using either inductive loops or any axle sensors.



Figure 2.24 Class 3 vs Class 5 (FHWA (Federal Highway Administration), 2014)



Figure 2.25 Class 3 vs Class 8 (FHWA (Federal Highway Administration), 2014)

The main differences across Classes 8, 9, and 10 lay in the number of axles that the truck-trailer combination has. Since inductive loop signatures do not have the ability to directly capture the axle number of each truck-trailer combination, correctly distinguishing these classes has been a challenge for signature-based models (Jeng et al., 2013). Nevertheless, the bagging DNN model significantly improved the performance of such classes over Jeng et al.'s approach (Jeng et al., 2013). However, misclassified vehicles were still observed among those classes due to the overlapping body type across FHWA classes. As Figure 2.26 shows, a Class 8 enclosed van (Figure 2.26a) was misclassified as a Class 9 (Figure 2.26b) vehicle, where Class 9 and Class 8 enclosed vans share similar shapes.

Likewise, the bagging DNN was found to misclassify a Class 10 drop frame van in Figure 2.27c into a Class 9, which is also a common axle configuration amongst drop-frame vans.

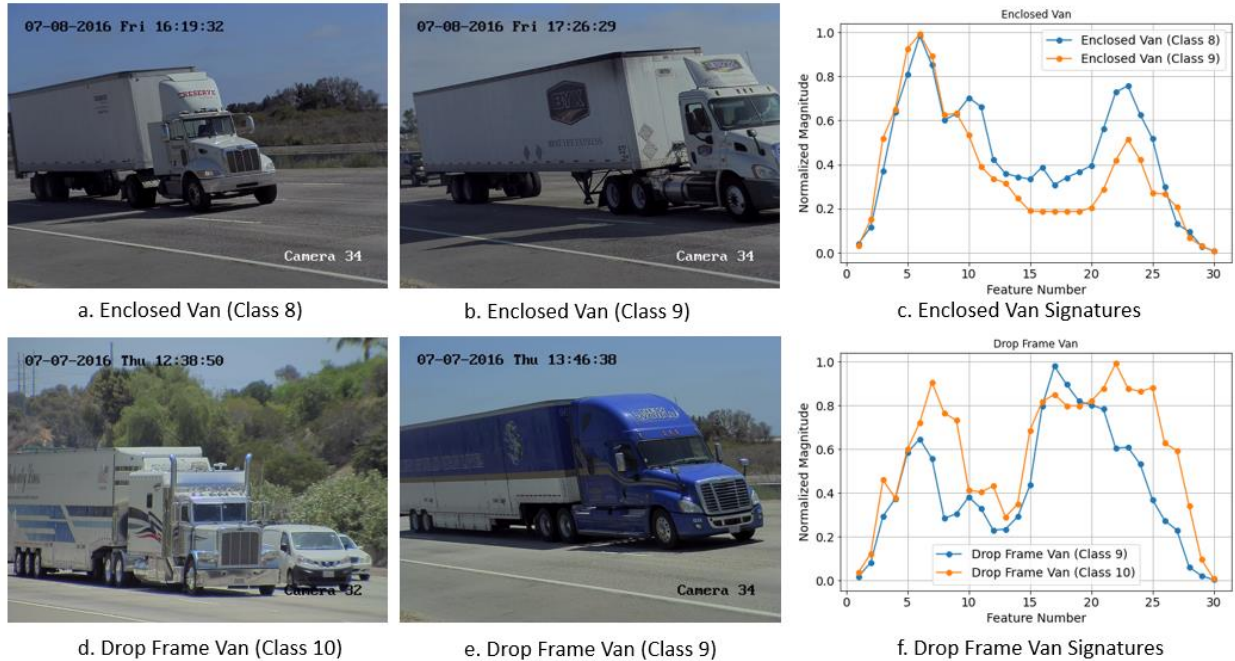


Figure 2.26 Overlapping body type across FHWA classes

Compared to conventional axle detectors, inductive loop signature data have demonstrated the ability to distinguish the body type of trucks with relatively high accuracy (Hernandez, 2016). Therefore, errors associated with overlapping axle configurations of different vehicle body types (refer to Figure 2.27) which are a major source of confusion at classification sites, were better managed by the signature-based bagging DNN model proposed in this study.

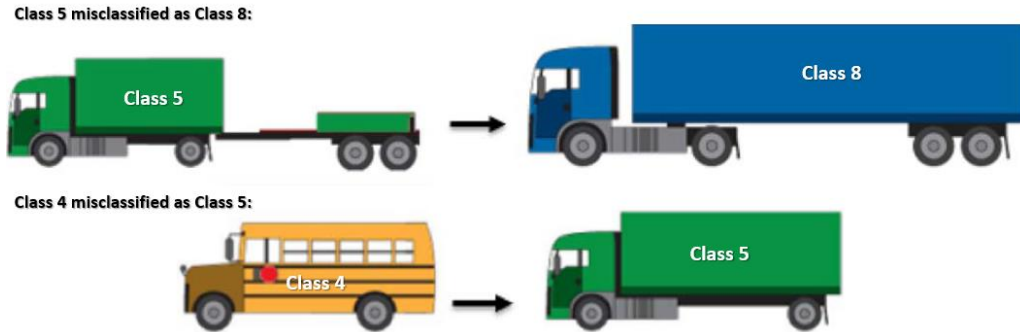


Figure 2.27 Error cases for piezoelectric sensors (Bitar & Refai, 2017)

## 2.2.6 Conclusion

This section proposed an accurate vehicle classification model to classify vehicles based on the FHWA-CA scheme using single inductive loops. The proposed model was developed on a truck-focused dataset and utilized the bagging ensemble technique to resolve the classification challenges of accommodating an imbalanced dataset which is typically observed in the FHWA classification scheme. The modeling process involved three major steps. First, the dataset was partitioned using the stratified sampling approach to retain a proportional number of samples from minority classes for both training and testing set. Then, a DNN model was constructed to assign signatures to their corresponding FHWA-CA classes. Dropout regularization was applied during the fine-tuning process, which successfully alleviated the overfitting of the DNN model. Finally, a bagging ensemble technique was used to address the imbalanced dataset issue. This bagging DNN model significantly outperformed a state-of-the-art FHWA classification model using inductive loop signature data (Jeng et al., 2013) for all truck-related classes. The bagging DNN model was able to achieve an F1 score of 0.83, where the comparable model obtained a value of 0.50. From the error analysis, it was observed that the majority of error cases came primarily from

the overlapping body types across FHWA classes. For instance, both Classes 8 and 9 shared body types such as semi-trailer enclosed van. In addition, semi-trailer drop-frame vans exist in both Classes 9 and 10. According to the error analysis, the proposed model likely inferred the FHWA classes through their corresponding body types. Notwithstanding, the overlapping axle configurations with different general body types, which is a common type of error at classification sites (Bitar & Refai, 2017) were still generally well-managed in the signature-based bagging DNN model.

Inductive loop sensors remain the most widely deployed detector infrastructure in the State of California and the United States, and inductive signature-based classification models have been widely deployed in the State of California (Tok et al., 2017). This research demonstrates that the improvements in the inductive signature-based model described in this paper are a cost-effective solution that can provide accurate classification performance across truck categories according to the FHWA scheme, while concurrently addressing common body configuration confusion issues experienced by axle-based detection systems.

Truck populations vary by state. The data used to train our model was obtained from several study sites across the state of California to capture a representative sample of trucks. However, some types of trucks found in the rest of the United States may not be represented in California. For example, Class 13 includes triple-trailer trucks that are permitted in many other states but are restricted from operation in California and were not included in our training dataset. The model can be further enhanced through training on a nationwide comprehensive truck signature dataset.

## **CHAPTER 3 : Non-Intrusive-Sensor Solutions: Rural Highways Truck Monitoring using Multi-array LiDAR Sensors**

### **3.1 Introduction**

Although inductive loop sensors are widely deployed in many urban and interstate highway corridors in the U.S., their coverage remains limited along rural highway corridors that also contribute significantly to the economy (Aschauer, 1990). However, the implementation of pavement intrusive sensors across the extensive rural highway network is impractical. Therefore, researchers have begun to investigate non-intrusive solutions for collecting vehicle classification data. The rapid advancement of Light Detection and Ranging (LiDAR) technology in recent years provides further opportunities for non-pavement intrusive alternatives to collect detailed vehicle classification data. In this chapter, a new truck classification method is developed using a LiDAR sensor array in a horizontal orientation, utilizing a reconstruction procedure that combines frames of sparse point clouds to generate a dense point cloud representation of vehicle objects to facilitate accurate truck classification, while preserving the panoramic LiDAR Detection Zone (LDZ). First, vehicle point clouds were extracted by removing the background and clustering the residual points into objects. Then, a new vehicle reconstruction framework was built to enrich the sparse point cloud obtained from the horizontally oriented sensor. Objects associated with the same vehicle from consecutive frames were grouped and combined to generate a dense 3D point cloud representation of each vehicle. Subsequently, the lower profile of the reconstructed vehicle point clouds was extracted and used to classify vehicles based on the FHWA-CA scheme. In contrast to previous studies, which used a classic machine learning framework, this chapter adopted the PointNet deep representation learning algorithm to train the classification model from the preprocessed point cloud data to classify trucks



according to their detailed body configurations. Both classification models are capable of producing promising prediction results.

### **3.2 Literature Review**

LiDAR technology was initially adopted for vehicle classification applications in the early 2000s when scanning laser sensors were available for the domain of traffic surveillance. Such sensors scan the cross-section of the roadway by taking several range measurements and generate gray-level intensity images for the vehicle passing through the scanning area. Abdelbaki et al (2001) used two laser scanners with a 10-degree separation to classify vehicles based on their aggregated bodies. In their study, high-level features such as vehicle length, vehicle width, vehicle height, and speed features were extracted from the intensity images and a rule-based classification lookup table was created to assign vehicles into their corresponding classes based on extracted features. Later, Hussain et al. (2005) adopted the same data collection setup with an additional feature, the average of the percent of edge points between two consecutive images, to further improve the classification accuracy. Instead of using a rule-based classification method, they constructed a random neural network model for the classification purpose. The prediction error was reduced with their improved classification method (Hussain & Moussa, 2005). Similarly, Sandhawalia et al. (2013) interpreted the 3D measurements acquired from a SICK laser scanner as a 2D image, where the pixel intensities were used as the depth values. The vehicle classification problem was posed as an image categorization problem. Instead of directly using the geometrical vehicle attributes from the profile image, Sandhawalia et al (2013) utilized the Fisher vector representation of the profile image, where a set of low-level local features obtained from the profile image was transformed into a high-level image representation. Subsequently, high-

level feature vectors were extracted from the fisher image signatures and were classified using the one-versus-all linear classifier. This model was able to classify vehicles into 6 categories with an average accuracy of 82.5 percent. The three aforementioned papers adopted overhead sensor mountings which offered the capability to capture detailed information on each passing vehicle. However, the overhead mountings were subject to infrastructure constraint. Therefore, researchers started to investigate using roadside LiDAR as a cost-efficient alternative for vehicle classification.

Lee and Coifman (2012) adopted side-fire LiDAR for vehicle classification. They designed a prototype data collection system consisting of a probe vehicle parked at the roadside, which was equipped with two vertically oriented laser scanners. Both sensors jointly scan the vertical planar of the road section. By merging successive 2D frames, a 3D LiDAR image can be constructed (Yang, 2009). After obtaining the 3D point cloud for the surveillance area, the vehicle objects were extracted from the background using a well-established background subtraction algorithm originated from the domain of image processing. Then, six high-level features describing the physical characteristic of each vehicle were extracted from the vehicle cluster as inputs to the classification model. Finally, using a decision tree classifier, the vehicle clusters were classified into six distinct classes refined from the length-based classification scheme. In addition, the researchers also explored using low-cost single-beam side-fire LiDAR to get truck body information. Asborno et al. (2019) grouped the raw distance measurements from the LiDAR sensor over time to build vehicle signatures and adopted a Bayesian combined predictor to classify trucks based on their aggregated body type classes. The roadside setup of the LiDAR sensors has been proved to be a cost-efficient and feasible solution for collecting classification data. However, neither

the laser scanner nor the single-beam LiDAR sensor is able to provide a detailed vehicle profile, which limited the classification accuracy and the total number of vehicles that can be classified.

In recent years, multi-array rotating 3D LiDAR sensors have become popular due to the sensing needs of autonomous vehicles. Nezafat et al (2019) mounted such a sensor on a roadside pole with a vertical orientation. The LiDAR orientation is illustrated in Figure 3.1.

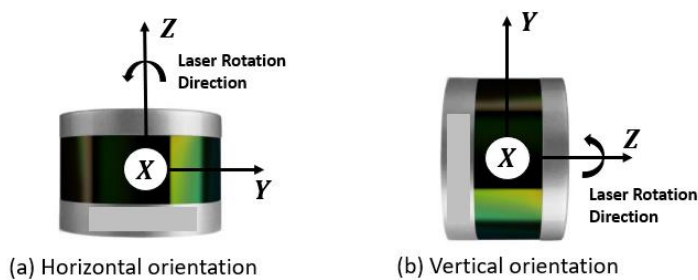


Figure 3.1 Illustration of LiDAR Orientation

When a truck entered the LiDAR detection zone, each scan of the sensor can capture a 3D profile of one side of the truck body within the scanning area. Due to the vertical orientation of the sensor, the detection zone was limited to 40 degrees of view (Figure 3.2).

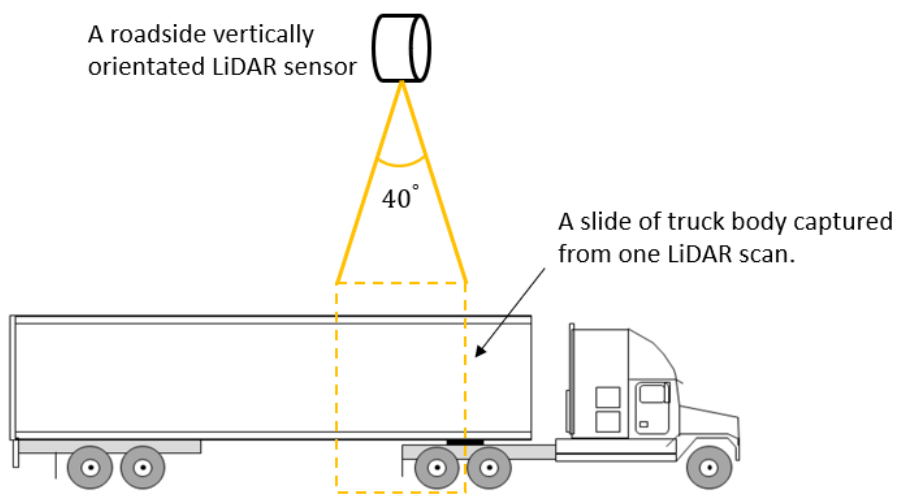


Figure 3.2 Truck Point Cloud Collection from Vertically Oriented LiDAR

Therefore, they had to merge all the frames associated with the same vehicle to generate the full profile of a truck. Then, they projected the 3D profiles of trucks to 2D images and adopted a pre-trained convolution neural network model to extract low-level features from the images. Such an image-based method was able to distinguish only 4 types of trucks with a similar configuration, yielding over 95% accuracy. However, the 3D information from the point cloud was not well-utilized in their research, which limited the total number of truck types that can be classified. Adopting the same data processing strategy with the vertical orientation of the sensor, Sahin et al (2020) utilized the 3D profile of the truck obtained from the merged frames to classify truck trailers in detail. Sahin et al. (2020) divided the 3D truck profile into six equal-sized rectangular voxels and extracted high-level features that represented the physical characteristics of different trailer types. Finally, five different classic machine learning algorithms were explored to classify 9 different trailers with the highest median accuracy of 94.2%. The vertical orientation of the multi-array rotating 3D LiDAR is able to capture a dense representation of each vehicle. However, the detection zone was narrow. It is hard to be extended for multi-lane traffic surveillance purposes. Wu et al. (2019) utilized a horizontally oriented LiDAR sensor, which provided a 360-degree view of the ambient environment, for vehicle classification. Unfortunately, the sparse point cloud representation retrieved from the horizontally oriented LiDAR gave insufficient information for detailed truck classification. Therefore, they were only able to distinguish three different types of trucks (Table 3.1). The evolution of LiDAR-based vehicle classification is listed in Table 3.1.

Table 3.1 Summary of LiDAR-based Vehicle Classification

Year	Literature	LiDAR Type	Setup	Methods Type	Classification Method	Correct Classification Rate
2001	Abdelbaki et al., (2001)	Laser scanner	Overhead mounted, two lasers	High-level Hand-designed features from Intensity Image	Rule-based Lookup table	Motorcycle: 66.6% Passenger vehicle: 87.2%; Pickup/Van/Sport Utility: 90.3%; Misc. Truck/Bus/RV: 84.7%; Tractor Trailer: 100.0%
2005	Hussain & Moussa, (2005)	Laser scanner	Overhead mounted, two lasers	High-level Hand-designed features from Intensity Image	Classic machine learning (random neural network)	Motorcycle: 60.0%; Passenger vehicle: 90.0%; Pickup/van: 94.4; Single unit truck or bus: 85.0%; Tractor Trailer: 100.0%
2013	Sandhawalia et al., (2013)	Laser scanner	Overhead mounted, each sensor per lane	High-level hand-designed features from raw profile feature, fisher image signatures, side projection profiles	Classic machine learning	Passenger vehicle: 99.8%; Passenger vehicle with one trailer: 89.8%; truck: 81.4%; truck with one trailer: 89.7%; truck with two trailers: 68.8%; motorcycle: 68.7%
2012	Lee & Coifman, (2012)	Laser scanner	Side-fire, vertically orientation, two lidars	High-level hand-designed feature from raw points	Classic machine learning	Motorcycle:91.2%; Passenger vehicle: 99.9; Passenger vehicle pulling trailer: 94.1%; Single unit truck: 94.5%; Single trailer: 68.9%; Multiunit truck: 98.6%
2019	Asbornio et al., (2019)	Single beam	Side-fire, horizontal orientation	Combination of High-level hand-designed feature and low-level feature from LiDAR signature	2D LiDAR signature pattern Classic machine learning (Bayesian combined predictor)	Van and container: 94%; Platform type: 63%; Low-profile trailer: 44%; Tank: 33%; Hopper and end dump: 30%
2019	Wu et al., (2019)	Multi-array rotating 3D LiDAR	Side-fire, horizontal orientation	Hand-designed features from raw points	Max height, the nearest distance to lidar, number of points in the frame, the difference between length and height, object profiles Classic machine learning	Bus: 100%; Five-axle, single-trailer truck: 94.1%; Bicycle; motorcycle: 5.9%; Three-axle, single-unit truck: 0%; Passenger car; four-tire, single unit; two-axle, six-tire, single-unit truck: 93.2%; Pedestrians and skateboarder:100%
2019	Vatani Nezafat et al., (2019)	Multi-array rotating 3D LiDAR	Side-fire, vertical orientation	Low-level feature from 2D images	Transfer Learning (AlexNet, VggNet and ResNet)	Container: 98.4%; Ref Container: 90.1%; Ref Enclosed Van: 95.7%; Enclosed Van: 97.6%
2020	Sahin et al., (2020)	Multi-array rotating 3D LiDAR	Side-fire, vertical orientation	Hand-designed features from vowelized point cloud	Classic machine learning	20ft Container: 96.3%; 40ft Container: 97.7%; 40ft reefer container:94%; Dry Van: 94.3%; Reefer dry Van: 91.0%; platform: 94.9%; Tank: 97.1%; Auto transport: 91.1; open top and dump: 85.1; other: 62.5

Regarding system setup, the LiDAR placement gradually transitioned from overhead mounting to cost-efficient roadside mounting. However, it is hard to capture dense point clouds from the side-fire setup. A vertically oriented LiDAR sensor is able to get dense point cloud while its field of view is very limited. On the other hand, horizontally oriented LiDAR

has a larger field of view, but the resulting point cloud is too sparse to generate useful information, which affected the accuracy of the classification model. Regarding methodologies, most of the methods in the literature are carried out in the same fashion. First, high-level features were selected and derived from either raw or transformed points, such as 2D depth images and voxels. Then, the high-level features were directly used as input for classic machine learning algorithms. The classic machine learning algorithms may work fine for the classification scheme with a small number of defined features (e.g., the axle-configuration in the FHWA classification scheme). However, for the task of body type classification problem, a combination of a myriad of features potentially can be used to differentiate various truck body types. Therefore, incorporating feature extraction into the optimization process of the classification algorithm would be much more ideal. In summary, it is challenging to collect enough information for truck classification while maintaining the view of detection. Therefore, in order to fill the gap, this chapter details a novel truck classification method, which presents a potential for multi-lane truck classification application and is capable of classifying based on both FHWA classification scheme and truck body types in detail with a promising classification result.

### **3.3 Data Description and Preprocessing**

#### **3.3.1 Study Site Layout**

The data used in this study were collected from the entrance ramp to the San Onofre truck scale from the Southbound I-5 Freeway in Southern California (as shown in Figure 3.3 Layout

of the Detection Site). Data under free flow and congested conditions were observed at the study site and included in the model development.

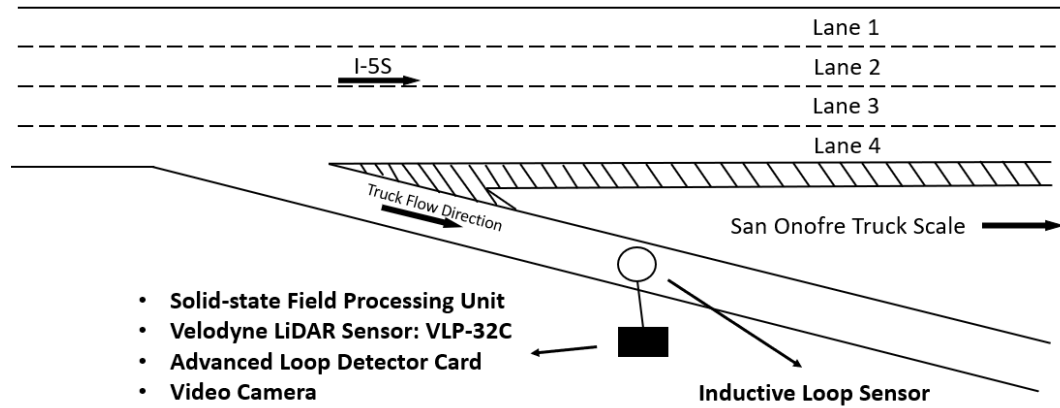


Figure 3.3 Layout of the Detection Site

### 3.3.2 Data Collection Setup

A video camera, an advanced loop detector card, and a Velodyne VLP-32c LiDAR unit were installed at the study site as shown in Figure 3.4. All three sensors were connected to a solid-state field processing unit. The video camera and loop detector were used to establish data ground truth. The Velodyne VLP-32c sensor has 32 infra-red lasers paired with infra-red detectors mounted on a motorized rotating platform to provide distance measurement between the sensor and objects (Velodyne Acoustics Inc., 2018).

The LiDAR was configured to scan the surroundings at a frequency of 10 rotations per second and 180-degree LDZ, with each rotation generating a single 3D point cloud frame. The LiDAR sensor was horizontally mounted on a platform attached to the roadside pole of an existing traffic control cabinet.

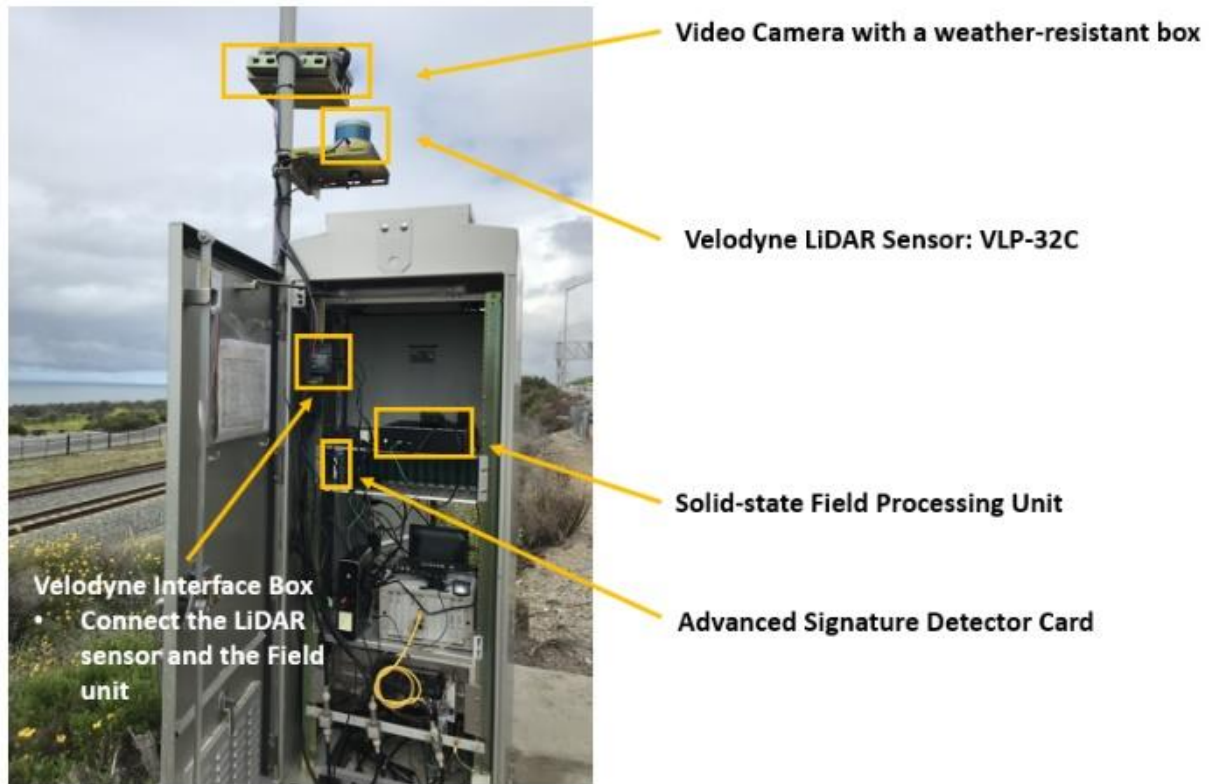


Figure 3.4 System Setup

The LiDAR sensor was mounted 2.05 meters above the ground plane and the top laser channel elevation angle was 15 degrees (Figure 3.5), which allowed the sensor to capture both the top and side view of passing vehicles. A sample of raw point cloud data frame showing a vehicle entering the truck scale within the LDZ is presented in Figure 3.6.



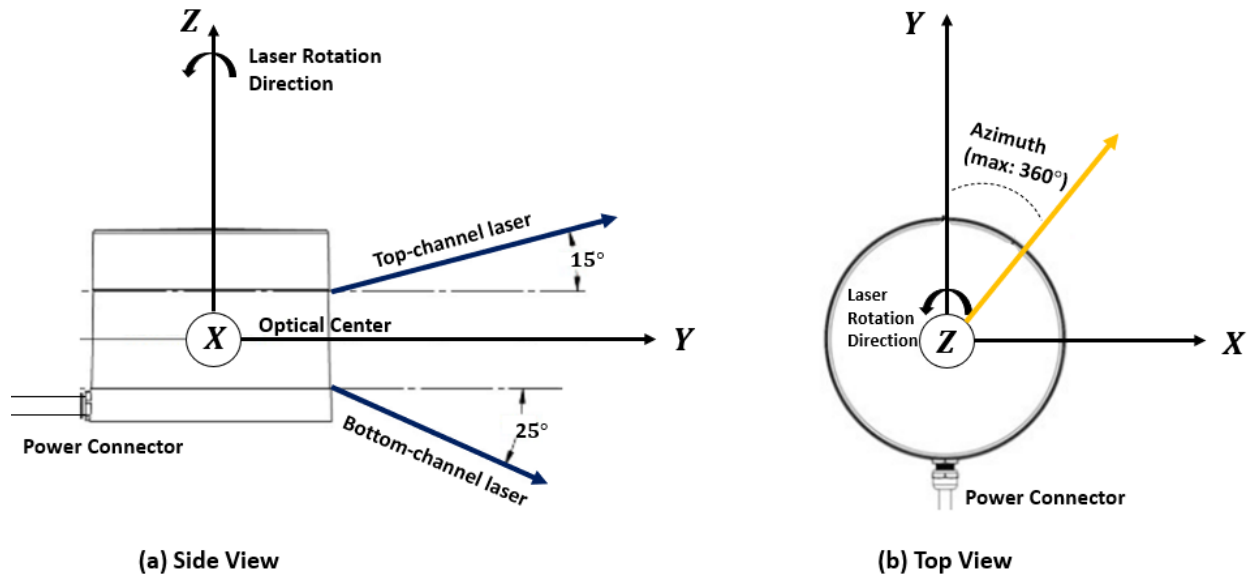


Figure 3.5 Illustration of the LiDAR Sensor

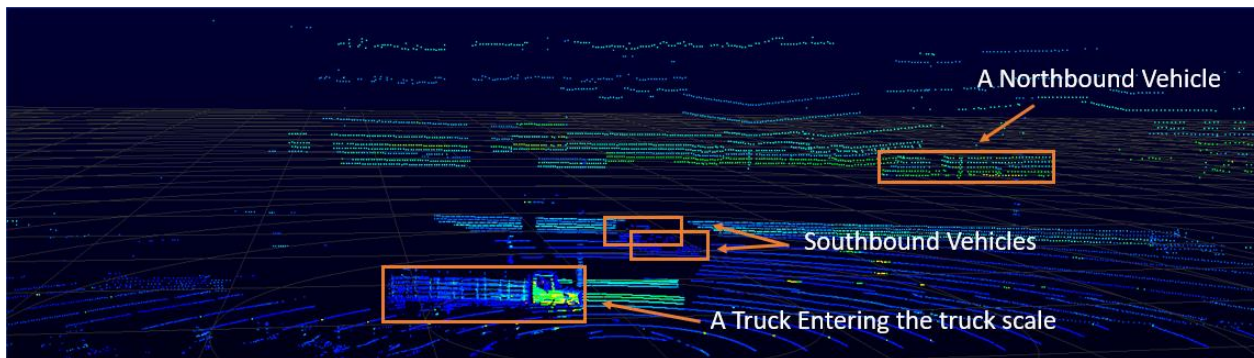


Figure 3.6 The Raw Point Cloud of the Detection region

### 3.3.3 Data Description

The video data from the camera, inductive loop signature data from the loop detector, and point cloud data from the LiDAR sensor, were collected simultaneously. The data collection started on July 18<sup>th</sup>, 2019, and ended on August 5<sup>th</sup>, 2019. Overall, point clouds associated with 10,024 vehicles were processed, representing 30 different types of trucks (including a class labeled as “Other” which represented trucks not belonging to any of the classes shown in Figure 3.7) as well as passenger vehicles (Figure 3.7). The vehicle point clouds have also

been labeled according to their FHWA-CA classes in this dataset. 70 percent of the data were used for training and 30 percent were reserved for testing.

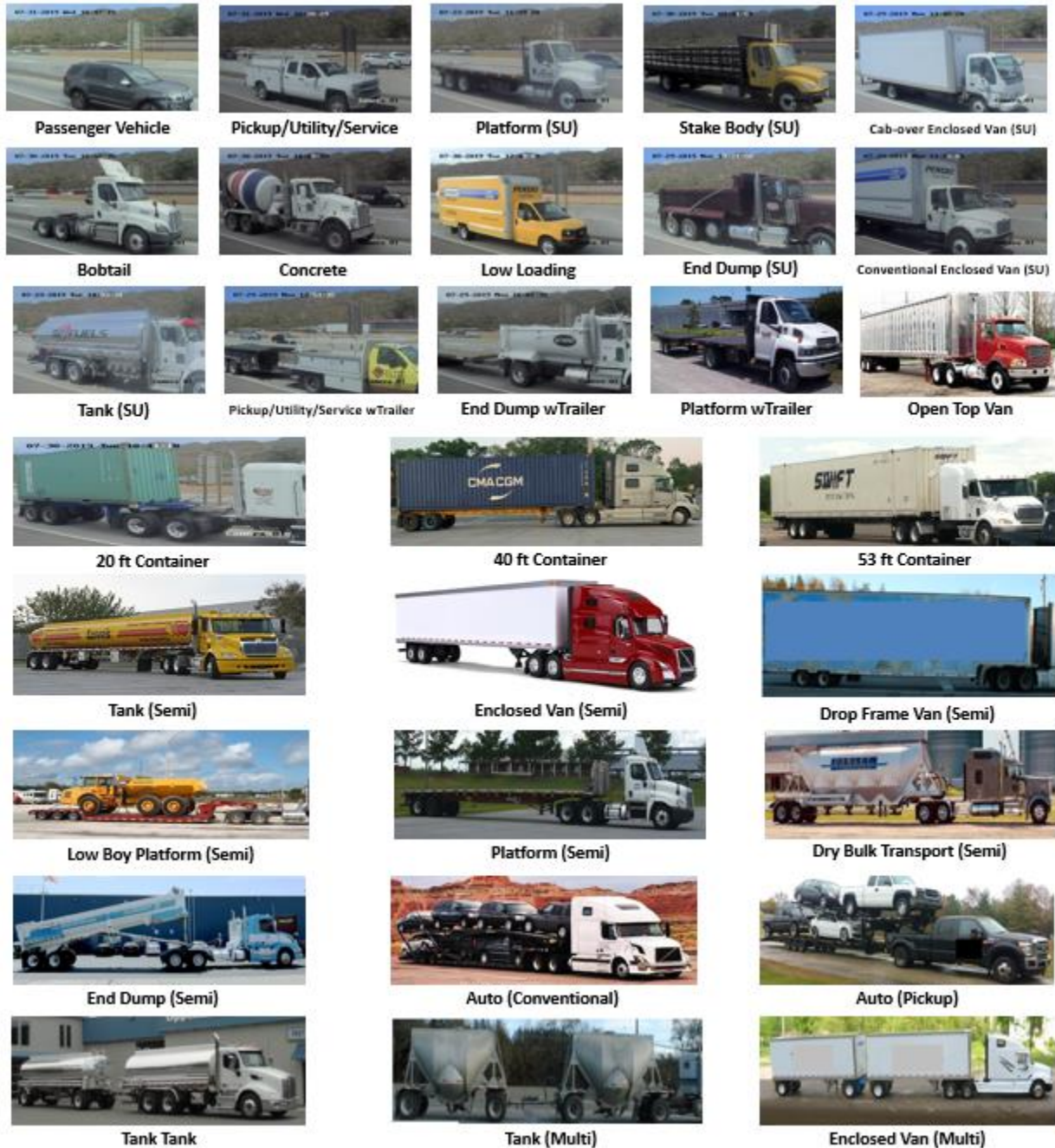


Figure 3.7 Illustration of Vehicle Body Configuration used in the study (Note: SU: Single Unit Trucks, Semi: Tractors pulling Semi-Trailer, Multi: Tractors pulling Multiple Trailers.)

### **3.3.4 Semi-automatic Data Labeling Method**

Data labeling has been considered a critical but labor-intensive process in many existing studies on vehicle classification. Conventionally, the data labeling process requires a large amount of time to visually verify the body configuration of each detected vehicle through images from the video camera, and then to manually record the vehicle characteristics correspondingly. In this study, a semi-automatic data labeling strategy was developed to improve the efficiency of the data labeling process and further enrich the training dataset. The overall semi-automatic data labeling process is illustrated in Figure 3.8. First, the vehicle records from three data sources were aligned. Then, this study adopted a signature-based truck classification model to pre-label vehicle images (Hernandez, 2014). Subsequently, misclassified data were manually corrected from visual validation of the pre-labeled dataset. Finally, the labels were applied to corresponding point clouds as the data sources were aligned. The signature-based truck classification model had an overall 72.4 percent accuracy (Hernandez, 2014). Therefore, the labels of only around 30 percent of vehicles needed to be corrected, which significantly reduced the workload of data processing. This semi-automatic data labeling strategy greatly accelerated the labeling process for emerging sensors (Sensor 2 in Figure 3.8) with the support of the existing technology (Sensor 1 in Figure 3.8).

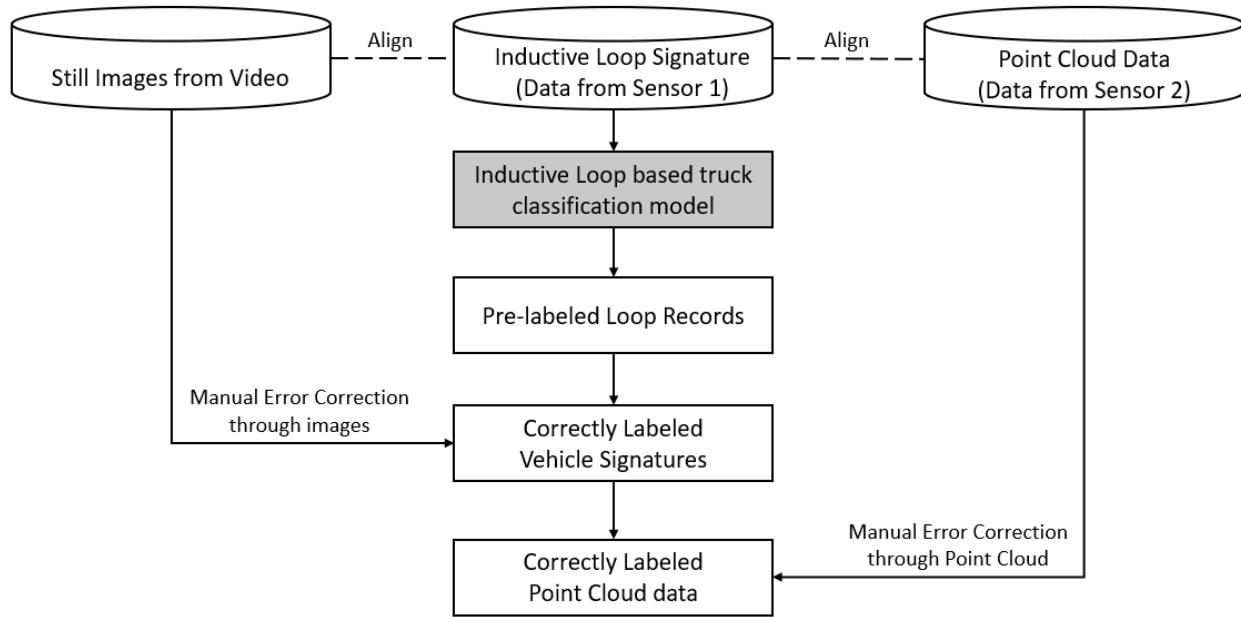


Figure 3.8 Semi-automatic Data Labeling Framework

### 3.3.5 Data Preprocessing

#### 3.3.5.1 Background Subtraction and Object Detection

As shown in Figure 3.9, each raw LiDAR scan contains both static roadway environment and the vehicle objects of interest. Prior to the modeling process, points belonging to each vehicle point cloud need to be segmented from the roadway background, which is irrelevant to the task of vehicle classification and are grouped into vehicle objects. This research adopted the background subtraction and object detection method proposed in (Li et al., 2021). The background subtraction method first divided the conical surface generated by the LiDAR sensor into annular sector-shaped cells. Then, the foreground vehicle point clouds and the background environment are split according to the spatial occupancy of each cell. Finally, the segmented vehicle point cloud was grouped and identified as a vehicle based on their points'

proximity using Density-based spatial clustering of applications with noise (DBSCAN) clustering algorithm (Daszykowski & Walczak, 1996).

### **3.3.5.2 Data Association**

The same vehicle object presents in multiple consecutive LiDAR frames. The vehicle object in each LiDAR frame needs to be labeled with the same vehicle ID. This research utilized the Simple Online and Realtime Tracking (SORT) algorithm to associate the vehicle point cloud from each LiDAR frame to its corresponding vehicle object efficiently (Bewley et al., 2016). First, each vehicle point cloud was represented by the centroid of the minimum oriented 2D bounding box which was obtained from its ground projection. Next, the inter-frame displacements of each vehicle point cloud were estimated using a linear constant velocity model—Kalman Filter (Kalman, 1960). Finally, the vehicle point clouds were optimally assigned to their corresponding vehicle object group using the Hungarian algorithm (Kuhn, 1955). SORT framework was claimed to be capable of handling short-term occlusion caused by passing objects (Bewley et al., 2016).

## **3.4 Vehicle Point Cloud Registration Framework**

### **3.4.1 Point Cloud Registration**

#### **3.4.1.1 Introduction to Point-Set Registration**

Point-set registration is an essential component that is widely used in the field of robotics and computer vision. It is the process of estimating the spatial transformation (e.g., translation, rotation, and scaling) that aligns two sets of points from the same object with a sensor that captures them from different views. Given two corresponding point sets  $P =$

$\{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \dots, \mathbf{p}_m\}$  and  $Q = \{\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \dots, \mathbf{q}_n\}$  in  $\mathbb{R}^d$  ( $d$  represents the dimension of each point. In this study,  $d = 3$ ), the goal of registration is to search for an optimal rigid transformation matrix  $\mathbf{T}_{PQ}$  composed of a rotation matrix  $\mathbf{R}(\theta_x, \theta_y, \theta_z)$  and a translation vector  $\mathbf{t}(t_x, t_y, t_z)$  to match point set  $P$  with point set  $Q$ . The parameters  $\theta_x, \theta_y, \theta_z$  represent the counter-clockwise rotation angle of the point set about the  $x, y, z$  axis, respectively. The values  $t_x, t_y, t_z$  are the translation of the point cloud along the corresponding axis. In a homogeneous coordinate, a transformation matrix  $\mathbf{T}_{PQ}$  that is used to align point set  $P$  and  $Q$  can be expressed as:

$$\mathbf{T}_{PQ} = \mathbf{T}_{PQ}(\theta_x, \theta_y, \theta_z, t_x, t_y, t_z) = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \quad (3.1)$$

The 3D rotation about  $x, y, z$  axis ( $\mathbf{R}_x, \mathbf{R}_y, \mathbf{R}_z$ ) and translation matrix  $\mathbf{T}$  is shown below:

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x & 0 \\ 0 & \sin\theta_x & \cos\theta_x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{R}_y = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{R}_z = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 & 0 \\ \sin\theta_z & \cos\theta_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

The most classic method used for solving point set registration problems is called the iterative closest point (ICP) algorithm (Besl & McKay, 1992). The ICP algorithm starts with the initial transformation matrix  $\mathbf{T}_0 = (\mathbf{R}_0, \mathbf{t}_0)$  and then selects a set of  $k$  corresponding points pairs  $(\mathbf{p}_i, \mathbf{q}_i)$  between point sets  $P$  and  $Q$ . The distance between  $P$  and  $Q$  can be written as:

$$\text{dist}(\mathbf{T}_{PQ}(P), Q) \quad (3.3)$$

$\mathbf{T}_{PQ}(P)$  represents rotating and translating  $P$  with a transformation matrix  $\mathbf{T}_{PQ}$ .  $\text{dist}()$  denotes the distance between point sets. In the literature, there are two common ways to

define the distance between point sets: Point-to-Point (Besl & McKay, 1992) and Point-to-Plane distance (Chen & Medioni, 1991).

### 1. Point-to-Point Distance Evaluation (Besl & McKay, 1992)

Assuming there are  $N$  corresponding point pairs  $(\mathbf{p}_i, \mathbf{q}_i), i = 1 \dots N$ , the registration problem using point-to-point distance measurement can be formulated as:

$$\operatorname{argmin}_{\mathbf{T}_{PQ}} \frac{1}{N} \sum_{i=1}^N \|\mathbf{T}_{PQ} \mathbf{p}_i - \mathbf{q}_i\|^2, \quad \text{s.t. } \mathbf{R}^T \mathbf{R} = \mathbf{I} \quad (3.4)$$

### 2. Point-to-Plane Distance Evaluation (Chen & Medioni, 1991).

When Point-to-Plane distances are used as the error metric, the objective function can be formulated as the sum of the square error between  $\mathbf{p}_i$  and the tangent plane at  $\mathbf{q}_i$ . The norm of the tangent plane at  $\mathbf{q}_i$  is denoted as  $\mathbf{nor}_i$ . The objective function is shown below:

$$\operatorname{argmin}_{\mathbf{T}_{PQ}} \frac{1}{N} \sum_{i=1}^N \|(\mathbf{T}_{PQ} \mathbf{p}_i - \mathbf{q}_i) \cdot \mathbf{nor}_i\|^2, \quad \text{s.t. } \mathbf{R}^T \mathbf{R} = \mathbf{I} \quad (3.5)$$

The next step of the ICP algorithm is to iteratively find the optimal  $\mathbf{T}_{PQ}$  which minimizes the distance between  $P$  and  $Q$ . Due to the simplicity of the original algorithm, hundreds of ICP-based variants have been proposed over the past two decades; a comprehensive review of ICP-based methods has been documented in (Pomerleau et al., 2015).

#### 3.4.1.2 Probabilistic Point-set Registration

However, the performance of ICP-based approaches suffers from the noisiness, outliers, and occlusions of point clouds that commonly occur in a real-world dataset, especially for data

collected from the outdoor environment (Pomerleau et al., 2013)(Myronenko & Song, 2010). In order to improve the robustness of point-set registration, many researchers started to investigate probabilistic approaches. The most popular probabilistic-based registration algorithm is called Coherent Point Drift (CPD) proposed by Myronenko and Song (2010), which treated registration as a probability density estimation problem. Instead of using the closest distance to define the corresponding point pairs, CPD assigned a probability value to the correspondence according to the proximity between points from two pointsets. Several studies followed the path of investigating new probabilistic approaches to further enhance the robustness of the registration algorithms (Evangelidis et al., 2014; Horaud et al., 2011; Jian et al., 2011). Unfortunately, such approaches gain robustness while compromising the computation efficiency, limiting their application to large datasets. Gao and Tedrake (2019) developed a computationally efficient probabilistic-based registration model—FilterReg—which adopted Gaussian filtering methods to enhance the model efficiency as well as to preserve the robustness and accuracy of the registration process. FilterReg has been proved to be computationally faster than the modern ICP variants (Gao & Tedrake, 2019). Therefore, this research adopted the FilterReg algorithm to estimate transformation matrices between consecutive frames.

### **3.4.2 Vehicle Point Cloud Registration Framework**

Most of the previous research on point-set registration targeted aligning point sets obtained from mobile sensors, where the LiDAR unit is mounted on the top of a moving robot (Pomerleau et al., 2015) which allows the sensor to actively capture the object point clouds. As a consequence, every point cloud density associated with the same object is relatively uniform across LiDAR frames. However, for traffic surveillance applications, LiDAR sensors



are generally mounted on a static pole standing by the roadside to passively monitor roadway traffic. As a vehicle traverses the LDZ, the density of its point cloud will gradually increase and then decrease due to its proximity to the sensor. Therefore, this study modified the existing point-set registration framework to better adapt to the data characteristics of vehicle point clouds collected from roadside LiDAR sensors, and then to provide promising vehicle point cloud registration results to support the needs of FHWA axle-based vehicle classification.

### 3.4.2.1 Eliminate Redundant Frames

When a truck is entering or leaving the LDZ (Figure 3.9), the position of the truck is far from the LiDAR sensing unit which caused the sparseness of the truck point cloud (Figure 3.10).

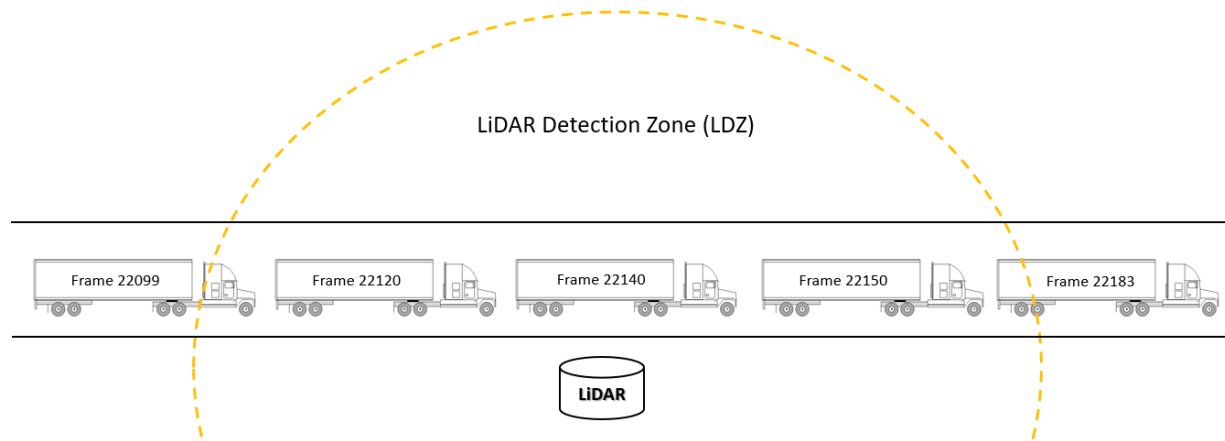


Figure 3.9 Samples of Truck Frames

Those frames generally describe the driving unit and the rear edge of the truck as shown in Figure 3.10.

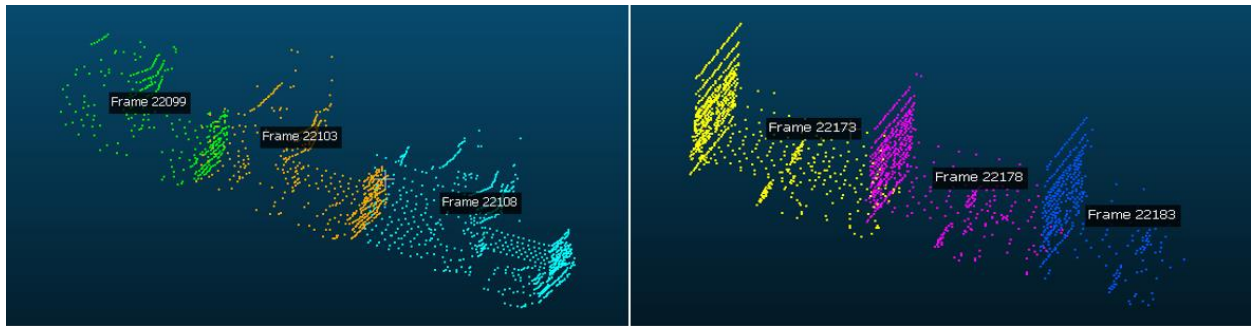


Figure 3.10 Samples of Redundant Frames

Such information has been included in the frames while the truck is closer to the sensor (Figure 3.11). Thus, those sparse point cloud which is captured far from the sensor and has limited contribution to the registration process were intended to be eliminated to save the computation time.

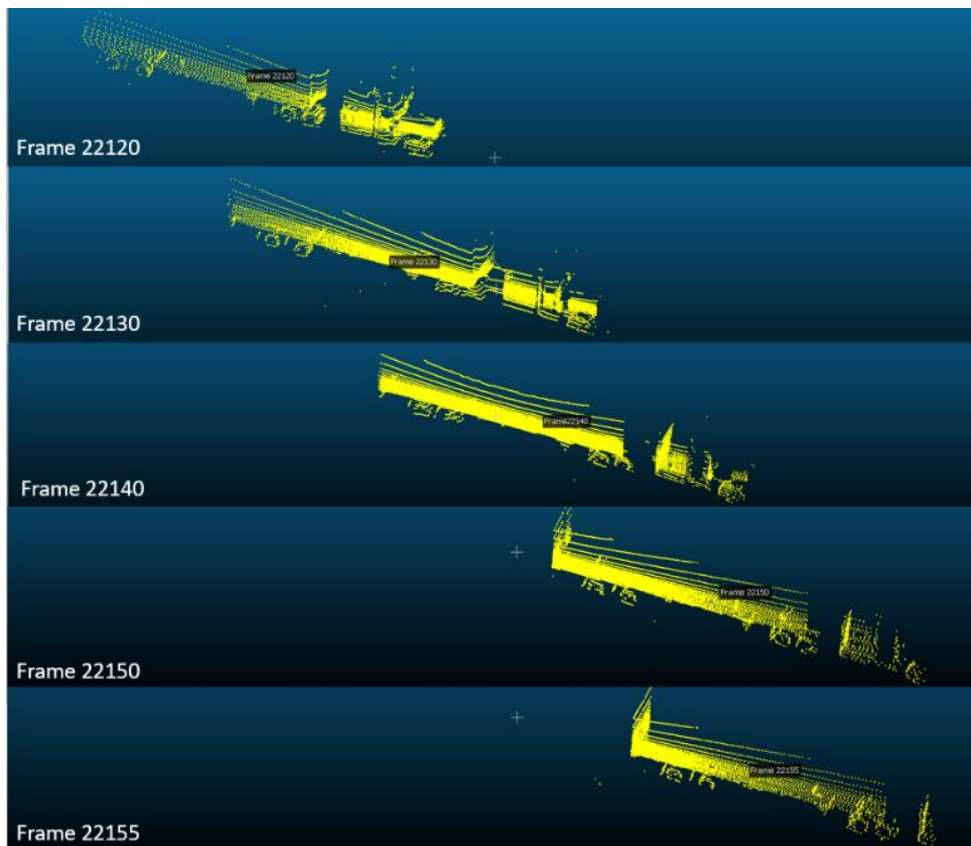


Figure 3.11 Samples of Frames used for Registration

Figure 3.12a presents the point counts profile while the truck is traversing the LDZ. Each point in the profile records the total number of points that the truck contained in its corresponding frame. Frame 22,138 and Frame 22,139 contain the highest number of points across all frames during its travel in the LDZ where the highest point count is denoted as  $pn_{max}$ . The point count profile is subsequently normalized based on  $pn_{max}$  (Figure 3.12b). Finally, the truck point cloud which contains point counts less than 20 percent of  $pn_{max}$  were treated as redundant frames and eliminated.

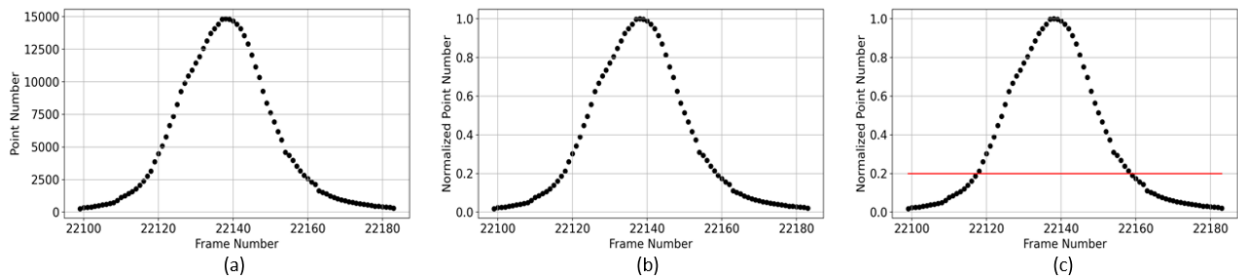


Figure 3.12 Elimination of Redundant Frames

### 3.4.2.2 Statistical Outlier Removal and Voxel Down Sampling

After the background subtraction step, there still existed noises and outliers which were statistically detectable. Therefore, prior to the vehicle point cloud registration, an outlier removal process is needed. In this step, two procedures are included, statistical outlier removal and voxel downsampling, which are suggested by a popular 3D data processing library—Open3D (Q. Y. Zhou et al., 2018). The statistical outlier removal method takes the 50 nearest neighbors of a given point in the point cloud and considers the points which are 2 standard deviations far from the given point as statistical outliers. Next, in order to increase the computational efficiency as well as preserve the structure of point clouds, point clouds are further uniformly downsampled using a voxel downsampling approach, where points

are bucketed into voxel with a size of  $vs\_pre = 0.01$  meter and represented by a single point calculated through averaging all points within the voxel.

### 3.4.2.3 Vehicle Point Cloud Registration

After the redundant frames and statistical outliers were removed, a pairwise registration with a coarse-to-fine strategy was applied on each pair of adjacent frames. The pairwise alignment was accomplished through the use of the FilterReg method (Gao & Tedrake, 2019). First, a coarse registration was conducted. All point clouds were coarsely downsampled with relatively larger voxel size  $vs\_coarse = 1.5$  meters and then each pair of point clouds was aligned based on the point-to-point distances metric. Transformation matrices were saved and denoted as  $\mathbf{T}_{j-1,j}^{coarse} = [T_{12}^{coarse}, T_{23}^{coarse}, T_{34}^{coarse}, \dots, T_{n-1,n}^{coarse}]$ , where  $j$  is the frame index. Second, the  $\mathbf{T}_{j-1,j}^{coarse}$  was fine-tuned using point-to-plane distances with voxel size  $vs\_fine = 0.015$  meter. The transformation matrices obtained from fine registration was written as  $\mathbf{T}_{j-1,j}^{fine} = [T_{12}^{fine}, T_{23}^{fine}, T_{34}^{fine}, \dots, T_{n-1,n}^{fine}]$ . Since the basic assumption about vehicle point clouds is that all the point clouds associated with the same vehicle should land on the same plane, the vehicles will not rotate along the  $x$  and  $y$ -axis. Hence, the transformation matrices were constrained on  $x$ - and  $y$ -axis rotation, where the corresponding elements in the matrices were set to zero, as shown in Equation 6:

$$\mathbf{T}_{j-1,j}^{ground} = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 & t_x \\ \sin\theta_z & \cos\theta_z & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.6)$$

Third, in order to reduce the cumulative errors which could be potentially caused by the sequential pairwise registration, the transformation matrices were further optimized using

the multiway registration which described a process of merging multiple frames of an object in a global space. In this study, multiway registration was implemented through the use of a pose graph optimization technique proposed in (Choi et al., 2015). The multiway registration process is illustrated as follows. First, the information matrices which represent the inverse correlation matrix between two consecutive transformation matrices were estimated. Second, a pose graph is defined with the transformation matrices ( $\mathbf{T}_{j-1,j}^{ground}$ ) as the node and information matrices ( $A_{n-1}$ ) as the edges in the graph, where each edge of the pose graph connects two nodes. The middle frame of the vehicle object was set to be the reference frame with index  $j = mid = ceil(\frac{n}{2}, 0.5)$ . All frames were aligned to the reference frame during the optimization process. The pose graph is optimized using the G2O graph optimization framework (Choi et al., 2015). The final transformation matrices that were used to reconstruct the vehicle point cloud were  $\mathbf{T}_{j-1,j}^{final} = [T_{12}^{final}, T_{23}^{final}, T_{34}^{final}, \dots, T_{n-1,n}^{final}]$ . The overall vehicle point cloud registration framework is shown in Figure 3.13.

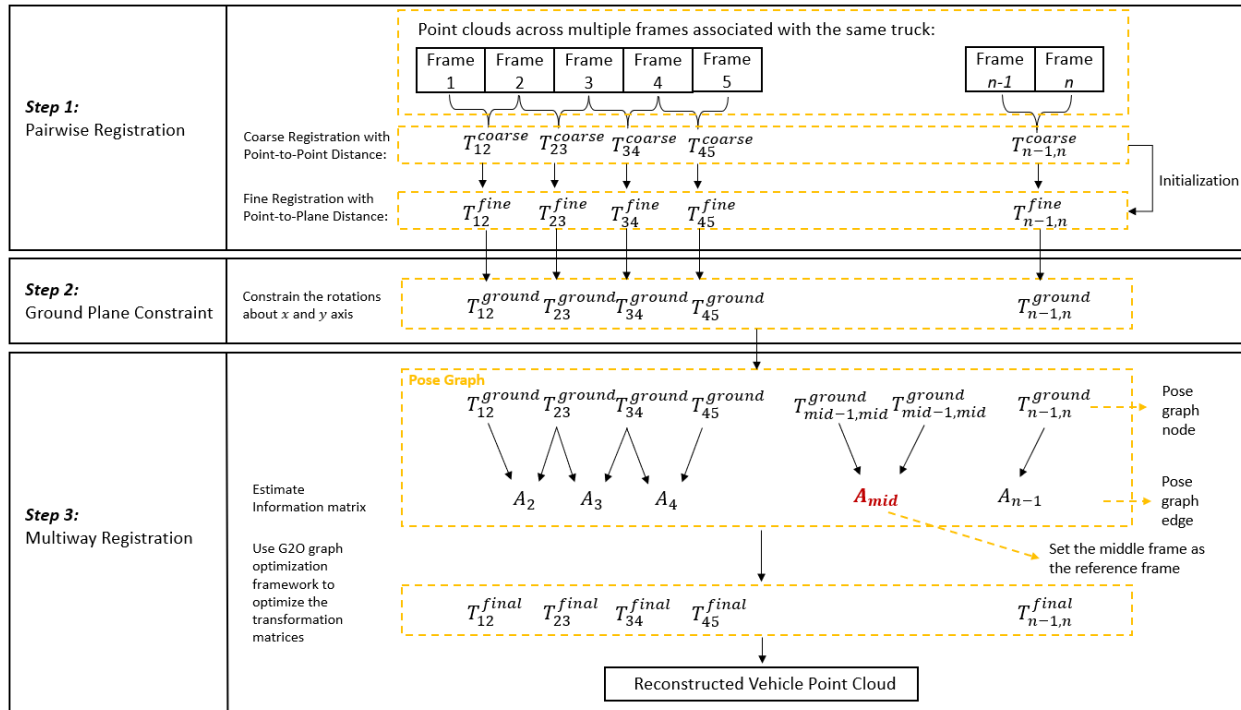


Figure 3.13 Vehicle Point Cloud Registration Framework

When the vehicle is approaching the LiDAR sensor, most of the information is captured from the tractor unit. The distinctive details as well as the level of the sparseness of the point cloud on the truck tractor make the process of finding corresponding points between two point clouds easier. Hence, minimizing the point-to-point distance is capable of aligning the source (Yellow in Figure 3.14) to the target point cloud (Blue in Figure 3.14) firmly. Figure 3.14a presents the point clouds from two consecutive frames. Figure 3.14b is the result of the coarse registration with point-to-point distance.

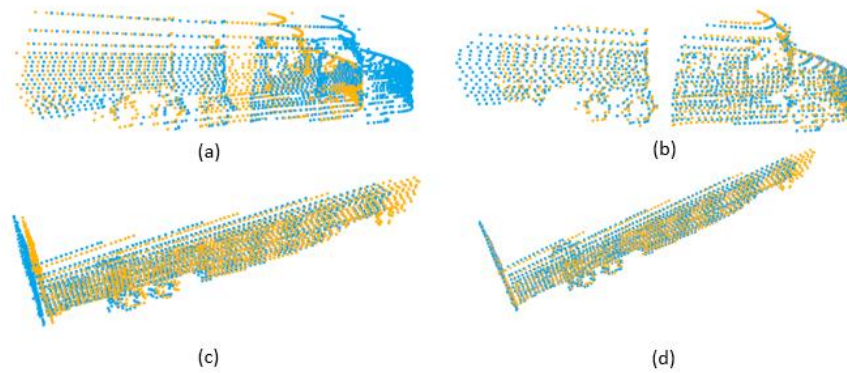


Figure 3.14 Examples of Pairwise Registration (Blue: target point cloud, Yellow: Source point cloud)

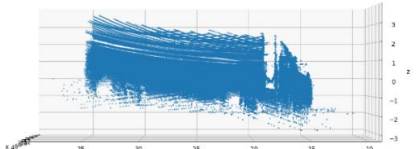
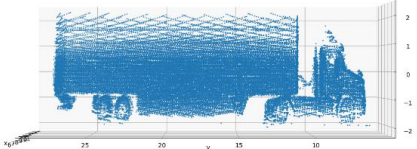
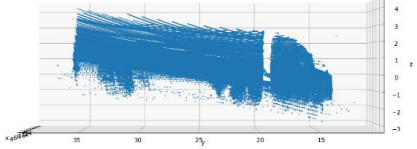
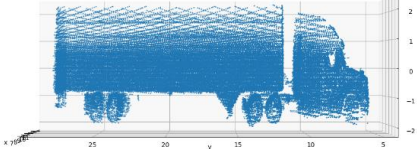
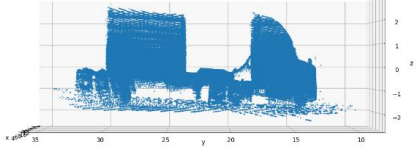
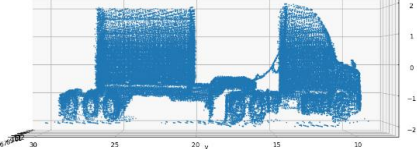
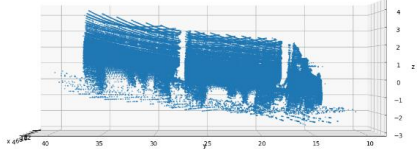
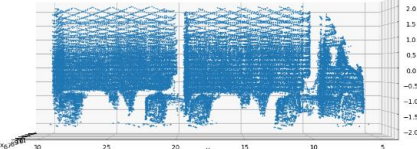
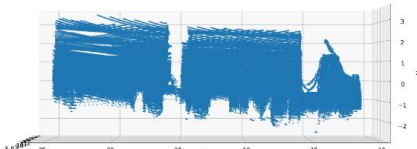
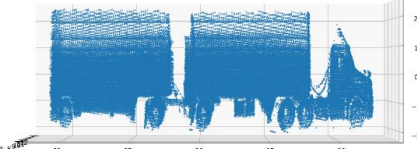
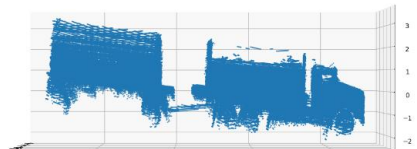
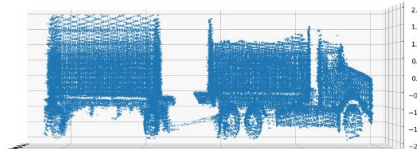
On the contrary, while the truck is just passed the LiDAR sensor, points are densely distributed on the side view of the truck. Such point clouds contained a limited number of prominent features to align them by just minimizing the point-to-point distance. Figure 3.14c shows the failure case after coarse registration using the point-to-point distance. However, the dense point distribution on the truck sides creates well-defined planes which allow the fine registration with a point-to-plane strategy to successfully further tighten two point clouds (Figure 3.14d).

#### 3.4.2.4 Registration Framework Comparison

The main purpose of the vehicle registration process is to enrich the information of the vehicle point clouds through merging multiple frames and to precisely portray the vehicle characteristics such that vehicles can be classified in detail. Therefore, if the reconstructed vehicle contains essential features which can be used to visually identify its vehicle class without any significant misalignment, it will be considered as a well-registered vehicle point cloud. Otherwise, it will be considered as a poor-registered vehicle point cloud. In this study, the vehicle point clouds registration precision (VPCRP) is defined as:

$$VPCR P = \frac{N_{wr}}{N_{wr} + N_{pr}}$$

Where  $N_{wr}$  represents the number of well-registered vehicle point clouds and  $N_{br}$  is the number of the poor-registered point clouds. The new framework has been compared with the previous registration framework through VPCR P value. The previous registration framework (Allu et al., 2020) presents a VPCR P value of 0.168, where the new registration framework has a VPCR P value of 0.024. The registration performance has been significantly improved by adopting the new framework proposed in this study.

FHWA-CA Classes	Previous Registration Framework	The New Registration Framework
Class 8		
Class 9		
Class 10		
Class 11		
Class 12		
Class 14		



### 3.4.2.4 Registration Performance with Missing Frames

The data used in this study was collected from the single-lane off-ramp area. Therefore, the occluded vehicle point clouds were barely observed in this data collection site. In order to test the robustness of the new registration framework, random frames for a truck object were dropped to simulate the missing frames scenario caused by either fully vehicle occlusions or sensor errors. Figure 3.15 demonstrates the experiment of the missing frame test.

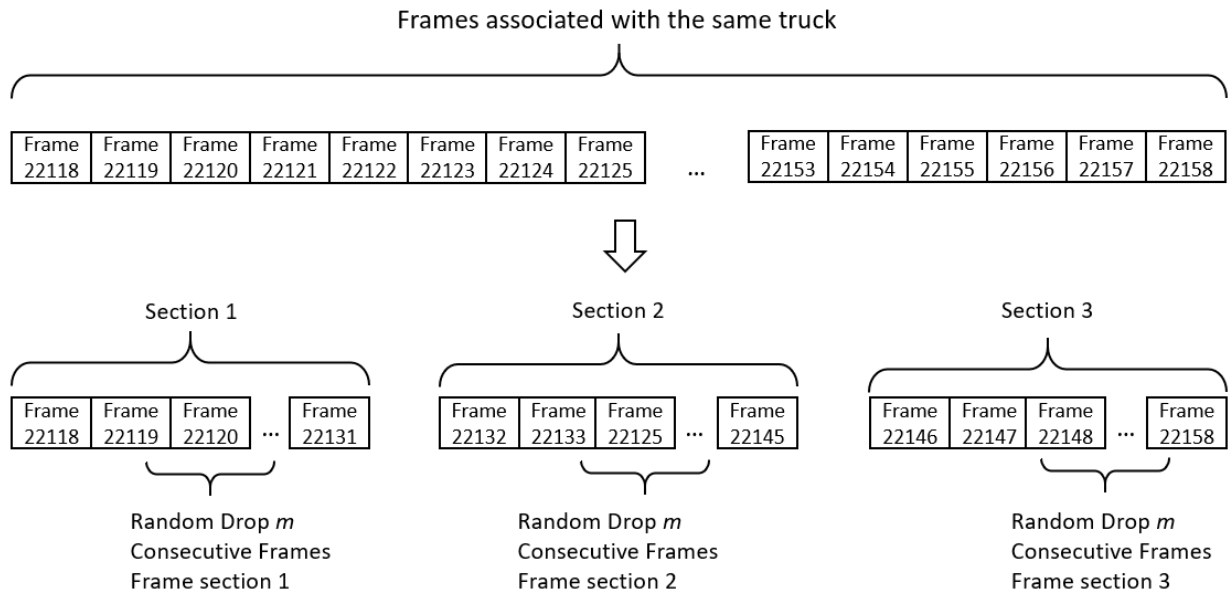
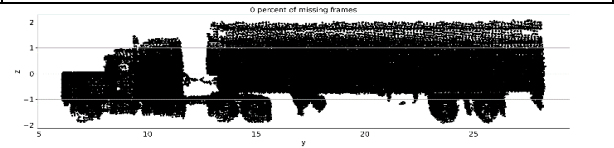
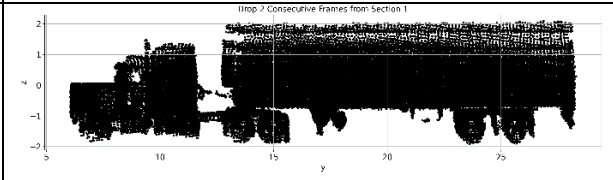
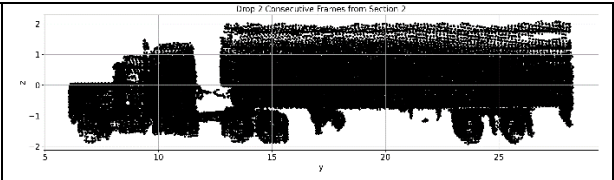
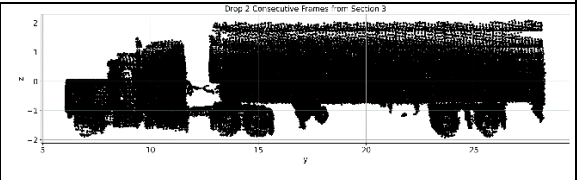
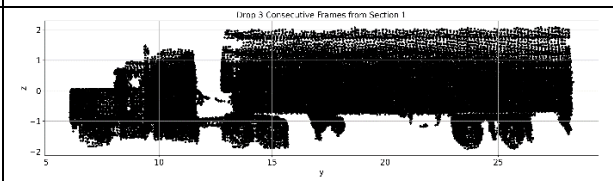
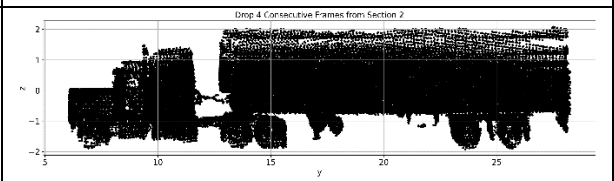
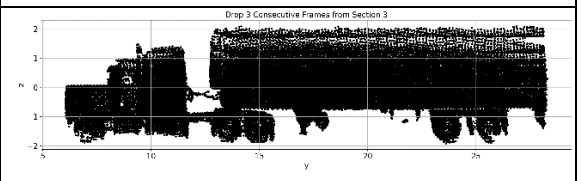
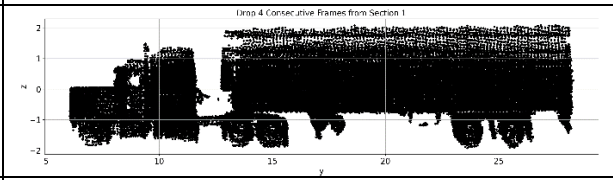
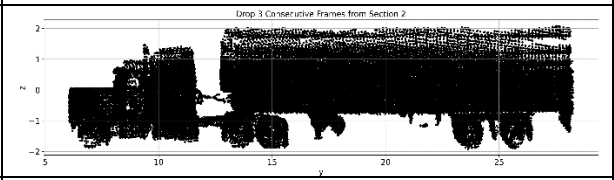
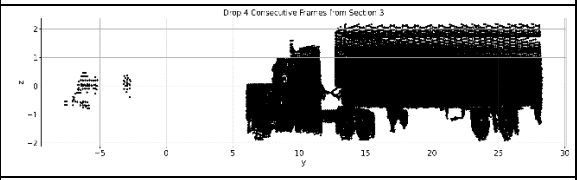
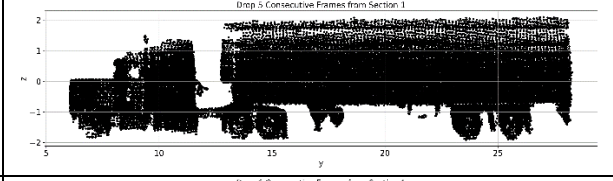
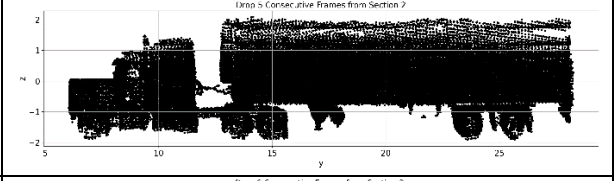
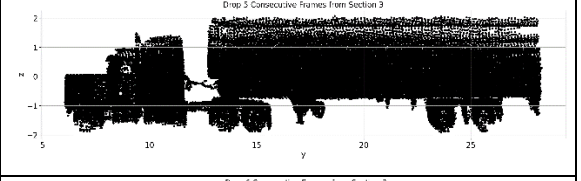
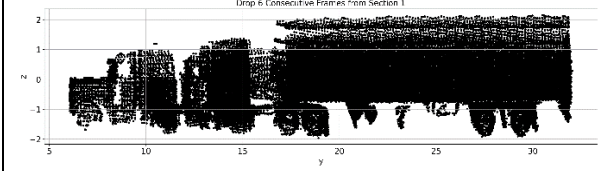
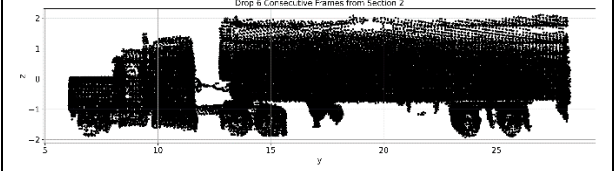
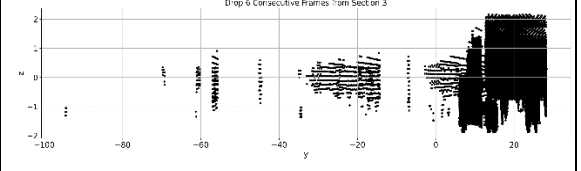


Figure 3.15 Illustration of Experiment Design

The duration that each vehicle traversing the LDZ was divided into three equal time slots denoted by Section 1, Section 2, and Section 3. Since vehicle occlusions generally happen on consecutive frames,  $m$  random consecutive frames are dropped from each section at each time.

Table 3.2 presents the results of the experiment.

Table 3.2 Experiment Results

Number of Consecutive Frames Dropped	Section 1: Approaching the LiDAR Sensor	Section 2: In front of the LiDAR Sensor	Section 3: Leaving the LiDAR Sensor
0			
2			
3			
4			
5			
6			
7	Completely miss aligned	Completely miss aligned	Completely miss aligned

When 5 consecutive frames are dropped, meaning that 0.5 seconds of data are missing, from either Section 1 or Section 2, the reconstructed point cloud is still able to preserve the essential information that can be used to identify their FHWA classes. For Section 3, the reconstruction framework fails when the consecutive frame number equal to 4. The random 5 consecutive frames dropped from Section 3 were the last 5 frames that are used for the vehicle reconstruction. Therefore, nearly a quarter of the points on the rear truck wheel were missing.

This experiment demonstrated that the proposed framework is capable of reconstructing vehicle objects with 3-5 consecutive missing frames. A comprehensive vehicle occlusion analysis will be explored after real-world occlusion data are collected.

### 3.5 FHWA Axle-based Classification using Roadside LiDAR Sensor

The lower profile of a truck contained information related to its axle and general body configuration which defines their FHWA-CA classes. Compared to each frame of a truck object, the lower profile of the reconstructed truck point cloud is well-defined (Figure 3.16).

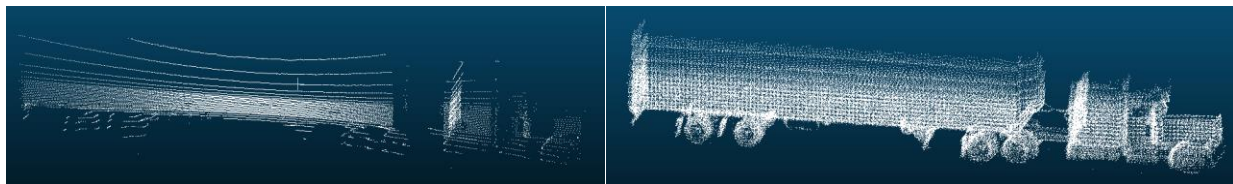


Figure 3.16 Truck Object from a Single Frame (Left) vs. A Reconstructed Truck Point Cloud (Right)

Therefore, in this section, essential features from the lower profile of the reconstructed truck point cloud were extracted and used as inputs for the vehicle classification model. Next, a deep ensembled neural network model was developed to assign vehicle point clouds to their corresponding FHWA-CA classes.

### 3.5.1 Feature Extraction

Prior to the feature extraction, statistical outliers on the reconstructed vehicle point cloud were further removed (Q. Y. Zhou et al., 2018). Subsequently, the pose of the vehicle point clouds was adjusted to align them with the zy plane using transformation matrix  $T_{mid-1,mid}$  since the middle frame was used as the reference frame in the pose graph optimization. Then, the 3D point cloud was projected to zy plane to obtain its 2D representation of each vehicle. The feature extraction process is shown Figure 3.17.

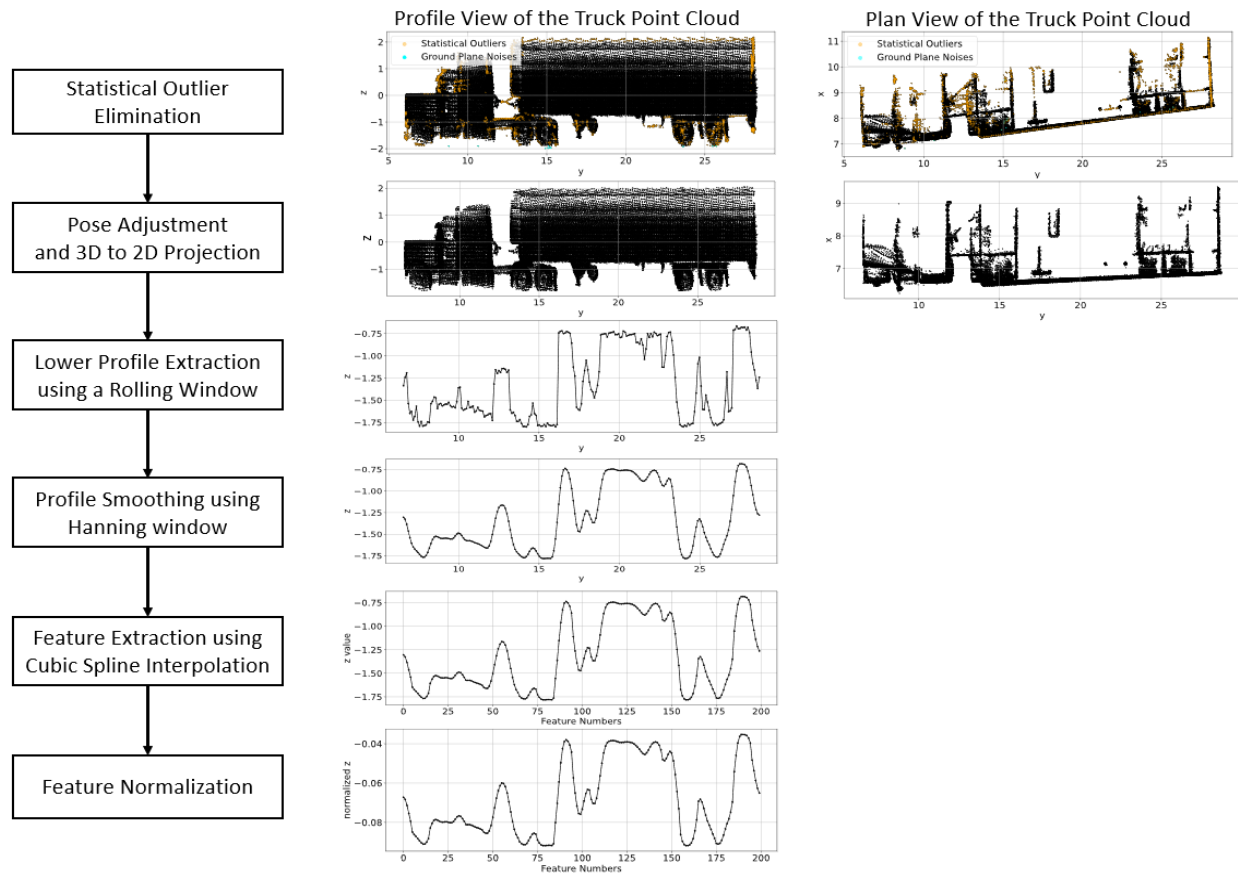


Figure 3.17 Feature Extraction

First, a rolling window with a size of 0.1 was created, where the minimum z value within the window was calculated. The size of the rolling window should be less than the radius of a regular wheel of a truck. The minimum z value rolling window captures the raw

lower profile of each vehicle point cloud. Second, in order to obtain a better representation of the lower profile, the raw profile was smoothed using Hann window (Oppenheim, V. et al., 1999) which is formulated as:

$$w(i) = 0.5 - 0.5 \cos\left(\frac{2\pi i}{M-1}\right) \quad 0 \leq i \leq M-1 \quad (7)$$

Where  $i$  represents the index of each point in the profile.  $M$  is the window size of the filter.

The smoothed lower profile of the truck point cloud presents both the axle and general body configuration of the truck. Third, the smoothed lower profiles were interpolated using cubic spline interpolation, and then 200 equally spaced  $z$  values were extracted from the interpolated profile to align the dimension of the training instances. Finally, the interpolated profile was normalized to the scale of -1 to 1.

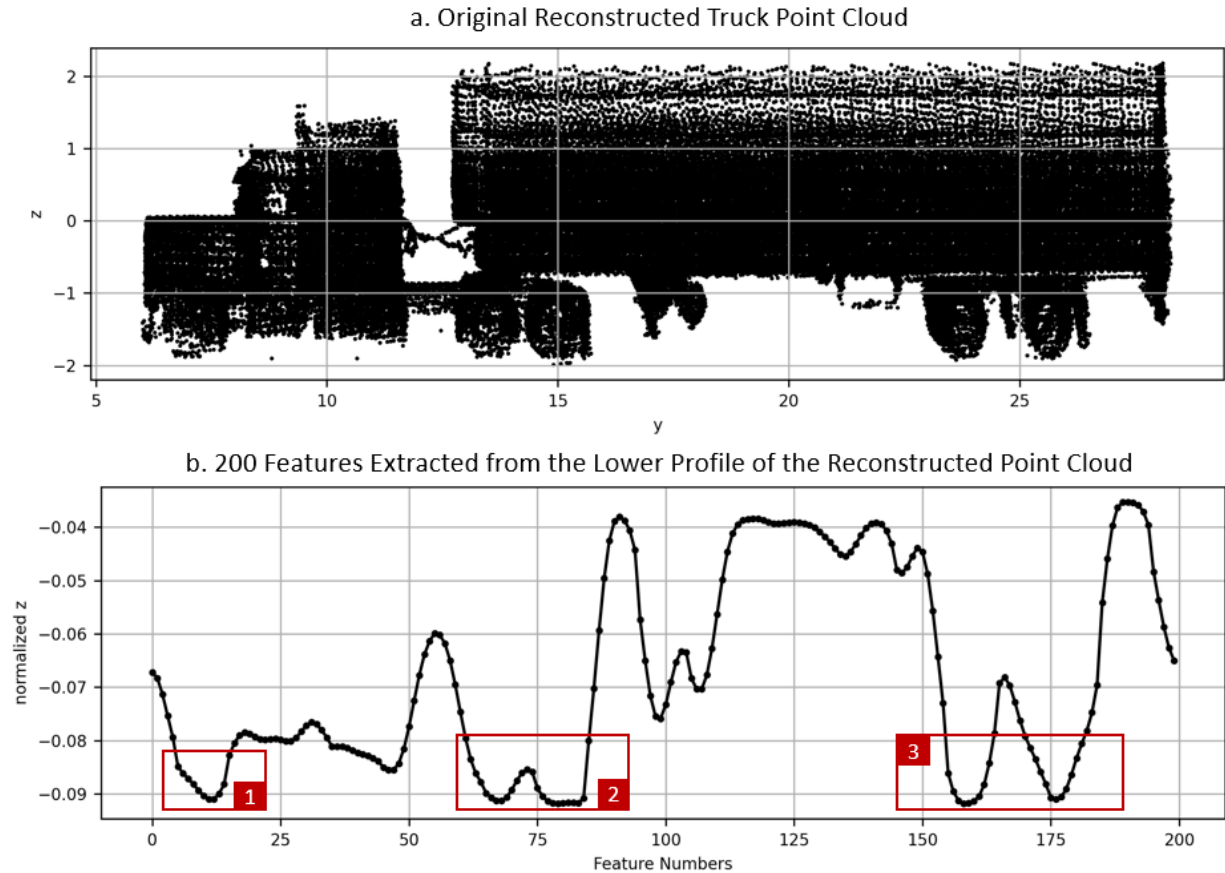


Figure 3.18 Illustration of Features

As Figure 3.18 shows, the valley in box 1 indicated the steer axle of the truck. Valleys in box 2 represent the drive axles on the tractor and the valleys in box 3 are the spread axles on the trailer. The peak shown in box 2 presents the connector between the tractor and trailer unit.

### 3.5.2 Bootstrap Aggregating Deep Neural Network for Vehicle Classification

Neural Network models have been proved to be able to approximate any complex non-linear mapping functions (Hornik et al., 1989). Compared to the shallow neural network, a multi-layer structure of a deep neural network model allows it to accomplish the same task with exponentially lower computation complexity (Shiyu Liang; R. Srikant, 2017). Therefore, this study developed a deep neural network (DNN) with dropout regularization (Mele & Altarelli, 2014) to assign each vehicle point cloud to its corresponding FHWA-CA classes. The DNN

model was composed of 5 hidden layers with 512 neurons on each layer. Thirty percent of neurons were randomly dropped out on the last two hidden layers to remedy the overfitting issue. The Rectified Linear Unit (ReLU) (Nair & Hinton, 2010) with the He initialization method (He et al., 2015) was applied to each hidden layer and the Softmax activation function with Xavier initialization (Glorot & Bengio, 2010) was used on the output layer. The learning curve shown in Figure 3.19 traces the model performance histories during the training and testing process. After 100 epochs, the overall accuracy on the training set keeps gradually increasing while the testing accuracy converges to 0.95. Hence the model training converged after 100 epochs.

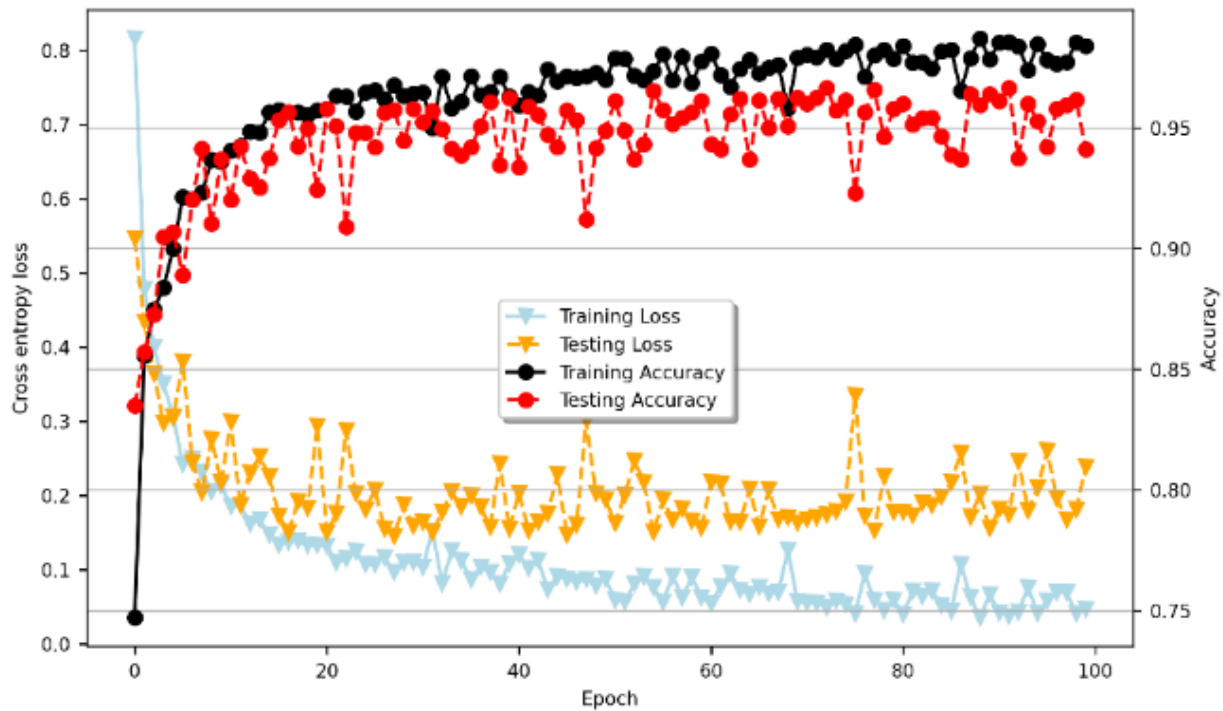


Figure 3.19 Learning Curve

In order to reduce the variability of the DNN prediction results, a bootstrap aggregating (bagging) (Breiman, 1994) ensemble approach was applied. In this study, the bagging ensemble method resamples the training set with stratified bootstrap resampling strategy

(Efron, 1988) to ten sets of bootstrapped training samples which were used to build ten different DNN models with the same model structure. The final prediction results were determined by the highest averaged prediction score of the ten DNN models.

### 3.5.3 Model Results

This section first evaluated the testing results of the proposed model using a normalized confusion matrix and then provide the error analysis on the misclassified vehicles. In addition, the proposed model was compared with the state-of-the-art FHWA axle-based classification model using a LiDAR sensor.

#### 3.5.2.1 Classification Results and Analysis

The normalized confusion matrix of the classification model is presented in Figure 3.20.

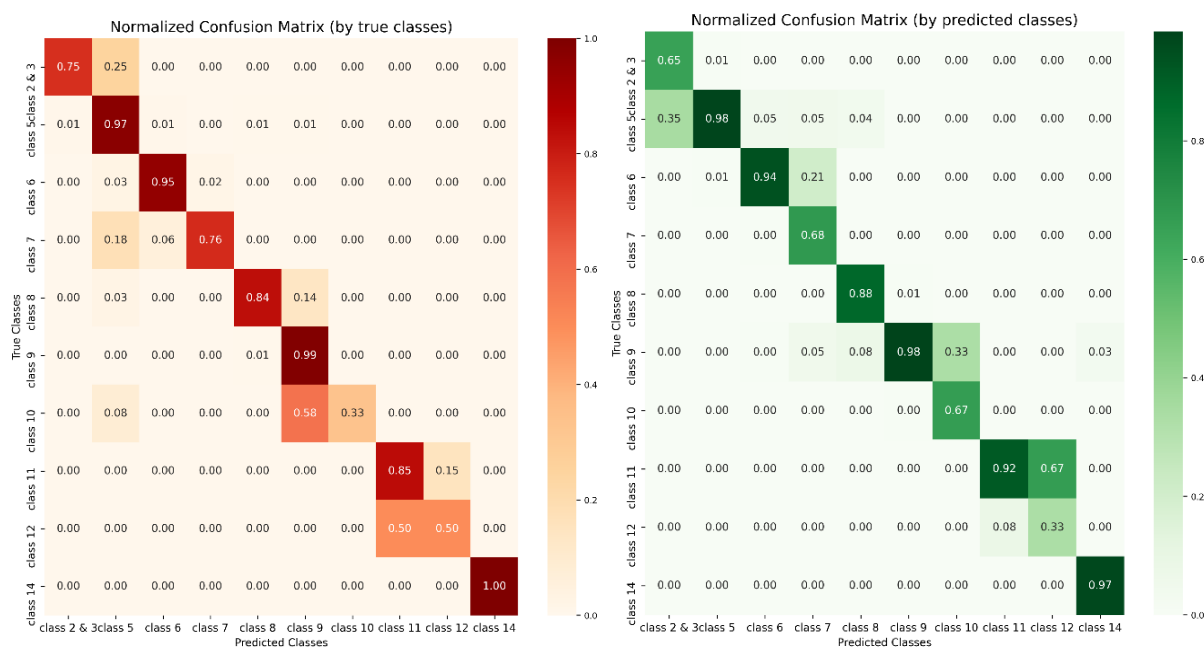


Figure 3.20 Normalized Confusion Matrix for the Test Set

Each row of the red-colored confusion matrix is normalized by a total number of ground-truthed vehicles in their corresponding classes. Therefore, the diagonal elements



represent the recall values of each class, which was also referred to as “Correct Classification Rate” (CCR) in some literature (Hernandez et al., 2016; Sahin et al., 2020). Each column of the green-colored confusion matrix is normalized according to the total number of predicted values for each class. Hence, the diagonal elements are the precision values of each class.

Based on the normalized confusion matrices, the proposed model is able to correctly classify Classes 5, 6, 8, 11, and 14 with over 80 percent CCR. However, the model is weak in predicting Class 10 and Class 12. Interestingly, in terms of Class 10, the precision value is higher than the recall value which means when this model is implemented, very few predictions on Class 10 will be received whereas most of them will be correctly classified. This is quite ideal for the model implementation. Conversely, Class 12 has a higher recall than its precision value which would cause that the model to return many Class 12 predictions, but most of them are misclassified from other classes.

The boxplot in Figure 3.21 shows the model recall distribution of the DNN models which are built with 10 sets of bootstrapped training instances. The bar plot represents the training sample size for each class.

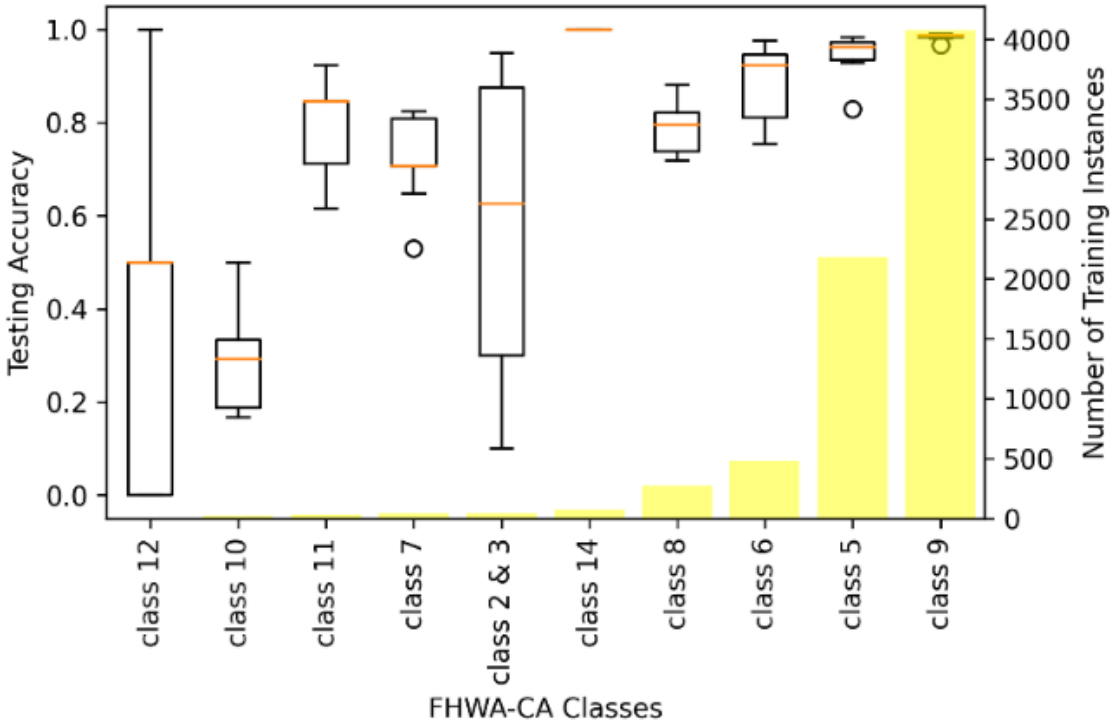


Figure 3.21 CCR Distribution across All Classes

As Figure 3.21 shows, the variability of the prediction results increases as the training sample size reduces, especially for Classes 10 and 12. Insufficient training samples were used to learn the key features from Classes 10 and 12 trucks which result in high variances in their prediction outcomes. In addition, Classes 2 and 3, passenger vehicles, are rarely observed at the entrance of the truck scale and those vehicles have larger diversity in terms of their body shape. Therefore, the model prediction variance is also high for Classes 2 and 3. Even though there is a limited number of training samples for Class 14, its prediction results are still promising since Class 14 represents a small homogeneous group of trucks.

With sufficient training samples, the proposed classification model is capable of accurately distinguishing Classes 8 and 9 with overlapping body configuration (Figure 3.22a and b). However, Classes which have minor differences in their axle configuration but with

the same body type are hard to recognize when the training instances are yet adequate (Figure 3.22c, d, e, and f). Consequently, in order to further enhance the model performance on Classes 10 and 12, the training dataset needs to be enriched in future studies.

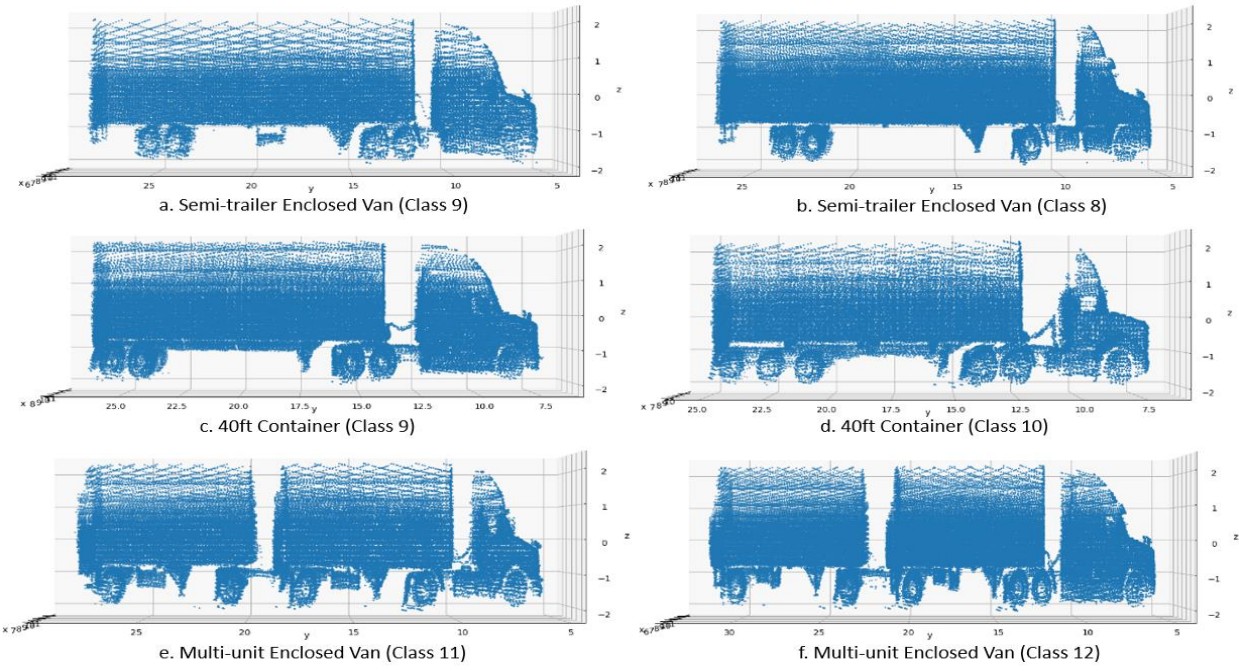


Figure 3.22 Overlapping Body Configurations

### 3.5.2.2 Model Comparison

The proposed model has been compared with the state-of-the-art LiDAR-based classification model which used the single frame of an object to classify vehicles on the basis of the FHWA scheme (Wu et al., 2019). The model comparison is shown in Table 3.3.

Table 3.3 Model Comparison

FHWA-CA	CCR (Bagging DNN)	Testing Samples	Classes defined in (Wu et al., 2019)	CCR (Random Forest) (Wu et al., 2019)	Testing Samples (Wu et al., 2019)
Class 2	0.75	20	Passenger Vehicle	0.84	150
Class 3			Four-tire Single Unit	0.70	69
Class 4	None	None	Bus	1.00	20
Class 5 <sup>1</sup>	0.97	934	Two-axle, six-tire, single-unit truck	0.44	17
Class 6	0.95	208	Three-axle, single-unit truck	0.00	4
Class 7	0.76	17	Four or fewer axle, single-trailer truck	None	None
Class 8	0.84	117	None	None	None
Class 9 <sup>2</sup>	0.99	1,746	Five-axle, single-trailer truck	1.00	17
Class 10	0.33	12	None	None	None
Class 11	0.85	13	None	None	None
Class 12	0.50	2	None	None	None
Class 13	None	None	None	None	None
Class 14	1.00	31	None	None	None
Average CCR	0.79	-	-	0.76	-

Note: <sup>1</sup>Class 5 used in this study contained a two-axle truck pulling a small trailer which was not included in (Wu et al., 2019). <sup>2</sup>In the FHWA-CA scheme, Class 9 type 32 was separated from the rest of Class 9 truck and labeled as Class 14. In (Wu et al., 2019), Class 14 trucks are merged into Class 9 trucks.

Compared to the previous model (Wu et al., 2019), the new classification framework proposed in this study is able to classify vehicles in much more detail with significantly higher accuracy, especially for heavy-duty trucks from Class 8 to Class 14 which have disproportionately adverse impacts on the pavement (Gillespie, T.D, Karamihas, S.M. & Sayer, 1993) and the environment (Guensler et al., 2005).

## **3.6 Truck Body Type Classification using Roadside LiDAR Sensor**

### **3.6.1 PointNet-based Truck Classification Model**

#### **3.6.1.1 The Deep Representation Learning Algorithm: PointNet**

The reconstructed 3D point cloud is an irregular type of geometric data structure, where each point is represented by its cartesian coordinates  $(x, y, z)$ . A conventional convolution neural network requires a regular data format such as image pixels and 3D voxels as inputs. Therefore, the point cloud cannot be directly fed into a typical convolutional architecture. Point clouds are generally transformed to other data types for classification purposes. In the literature, transportation researchers usually extract high-level physical characteristics from either raw points (Wu et al., 2019) or transformed point clouds, e.g., 2D images (Abdelbaki et al., 2001; Hussain & Moussa, 2005; Sandhawalia et al., 2013; Vatani Nezafat et al., 2019) or 3D voxels (Sahin et al., 2020) to solve the truck classification problem. Finally, these high-level features are used as inputs of classic machine learning algorithms. However, such data transformations and aggregations can introduce quantization error and further conceal the natural invariances of the point cloud data (Qi et al., 2017), which affects the accuracy and the variety of types of trucks that can be classified. In order to accommodate the characteristics of the point cloud data structure for improving truck classification accuracy, a novel deep neural network architecture—PointNet (Qi et al., 2017)—was adopted in this research. This neural network architecture can directly take point clouds as inputs and detect critical features for classification from the raw inputs. The architecture of PointNet is shown in Figure 3.23.

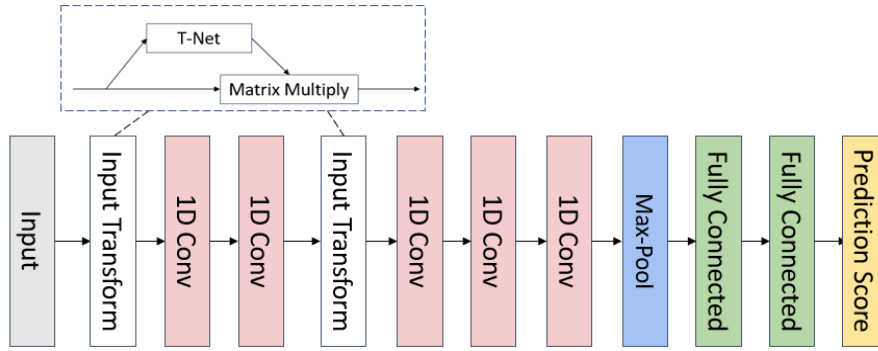


Figure 3.23 PointNet Architecture (Qi et al., 2017)

PointNet primarily benefits from two components of its architecture: the shared multi-layer perceptron (MLP) and the max-pooling function. The shared MLP was constructed using 1D convolution with a kernel size of 1, which provides a dense connection across points with the shared parameters (weight and bias terms). This means that the spatial encoding of each point can be learned by the shared MLP. A max-pooling layer was applied as a symmetric function to gather information from all the points, in order to resolve the invariance to permutation issue of the point cloud data structure. A function  $f(x_1, x_2, \dots, x_N)$  of  $N$  variables are invariant under random permutation if the function value does not change over the permutation of its variables. The generic representation of symmetric functions can be written as:

$$f(x_1, x_2, \dots, x_N) = f(x_N, x_2, \dots, x_1) = f(x_2, x_1, \dots, x_N) = \dots \quad (3.1)$$

The max-pooling function extracted the global critical feature of each truck point cloud and the overall model structure was able to learn the skeleton of each object. Since the truck body types are generally invariant and distinct in shape, PointNet ideally fits the task of truck body classification.

### 3.6.1.2 PointNet for Truck Classification Model

In this research, the PointNet architecture was adopted to classify truck body types in detail. Prior to the training process, the reconstructed point cloud needed to be regularized. First, the variable number of data points in reconstructed point clouds was uniformly downsampled to a common number of points as inputs into the PointNet. The downsampling process contained three steps. First, a regular voxel grid with a resolution of 5 percent was generated for each reconstructed truck point cloud, where those points were bucketed into voxels. Second, each occupied voxel was represented by a single point, which was calculated by taking the average of all points within each voxel grid. Finally, 1024 points suggested by Qi et al. (2017) were randomly sampled from the uniformly downsampled point cloud. A truck point cloud  $k$  can be written as a 3D point set,  $p^k = \{(x_j^k, y_j^k, z_j^k) | j = 1, \dots, n\}$ , where  $n = 1024$  in this study. After the downsampling process, the centroid of the truck point cloud  $k$  was moved to the (0,0,0) point in the coordinate and was represented as  $p^{kc} = \{(x_j^{kc}, y_j^{kc}, z_j^{kc}) | j = 1, \dots, n\}$ . The operation along the x axis is presented in Equation 3.2, where y and z follow the same calculation.

$$x_j^{kc} = x_j^k - \frac{\max\{x_j^k\} - \min\{x_j^k\}}{2} \quad (3.2)$$

Then, the truck point cloud  $k$  was normalized to a unit sphere and denoted by  $p^{kcn} = \{(x_j^{kcs}, y_j^{kcs}, z_j^{kcs}) | j = 1, \dots, n\}$ . The operation along the x-axis is presented in Equation 3.3, where y and z follow the same calculation.

$$x_j^{kcs} = \frac{x_j^{kc}}{\max\{x_j^{kc}\}} \quad (3.3)$$

The point cloud preparation step is shown in Figure 3.24. Here, a reconstructed point cloud of an auto transport with a conventional tractor is taken as an example.

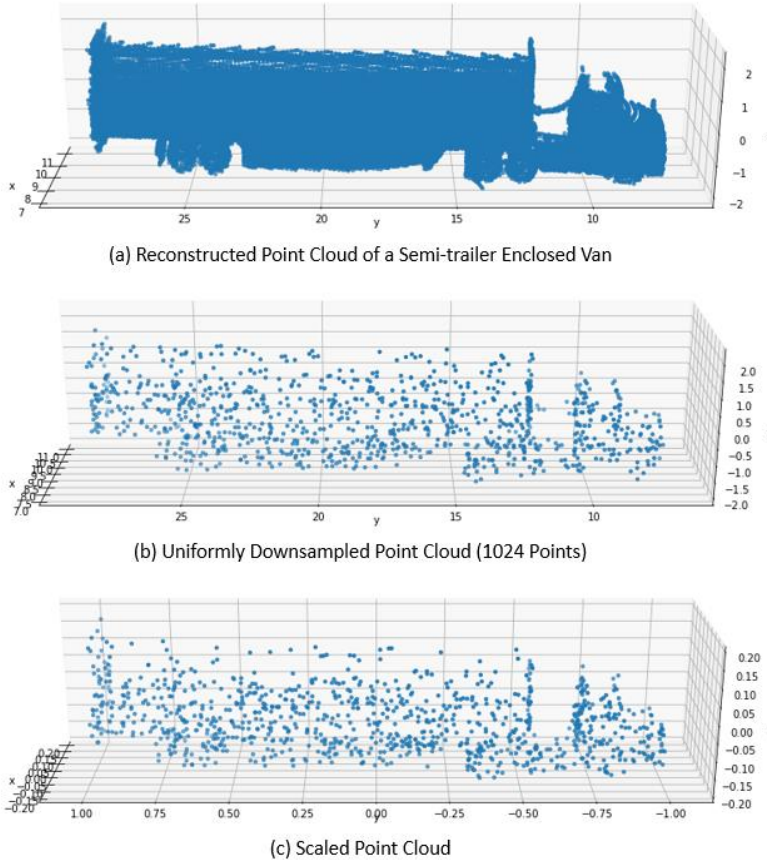


Figure 3.24 Point Cloud Preprocessing

During the model training process, two data augmentation methods were applied (Qi et al., 2017). First, each training instance was randomly rotated along the z-axis. Second, each point of the truck point cloud was jittered with a Gaussian noise which followed a  $N(0, 0.02)$  distribution to increase the diversity of the training instances.

The truck classification model was trained on 5,360 reconstructed truck point clouds with an RTX 2080 super GPU and took approximately 3 hours to converge. The learning process of the truck classification model is presented in the learning curve (Figure 3.25). The model accuracy on both training and test dataset improve in a similar trend until 100 epochs.



After this point, the model performance gradually plateaus on the test dataset but continues improving on the training dataset. The model converged after 250 epochs.

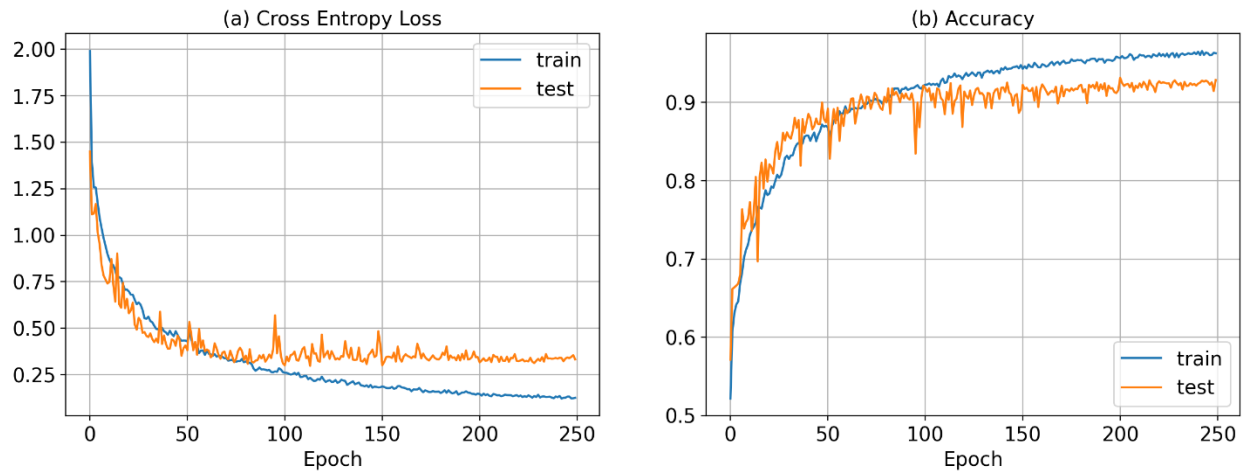


Figure 3.25 Learning Curves

### 3.6.2 Model Averaging

A multiple layer structure with nonlinear activation functions on each layer provides deep neural networks with the ability to approximate any complex mapping function (Goodfellow et al., 2016). However, deep neural network models generally suffer from high variance issues, where model performance varies significantly by dataset (Z. H. Zhou et al., 2002). Hence, model averaging strategies were explored to reduce the model variance and further enhance the model performance. The simplest way to apply model averaging on deep neural networks is to train multiple deep neural networks with different initial values and have all the models cast their votes. In this study, two model averaging methods were explored, and are explained in the next subsection.

### 3.6.2.1 Simple Model Averaging (SMA)

Let  $m_a = \{m_1, m_2, \dots, m_n\}$  denote  $n$  PointNet models trained with various initial values.  $c_b$  denotes the class labels.  $p(c_b|m_a)$  represents the probability that model  $m_a$  predicted class  $c_b$ . The equation of SMA is shown below.

$$\hat{c} = \underset{c_b \in \mathcal{C}}{\operatorname{argmax}} \sum_{a=1}^n p(c_b|m_a) \quad (3.4)$$

SMA assumes that  $m_a$  produced an equal contribution to the final decision and gave the prediction results by averaging all the votes of the candidate models.

### 3.6.2.2 Bayesian Model Averaging (BMA)

Unlike simple model averaging, which treats candidate models  $m_a$  equally, Bayesian model averaging assign a prior probability, presenting the subjective credibility of the model predicting a certain class. The posterior probability derived from the candidate models was used as the final prediction score (Hoeting et al., 1999; Raftery et al., 2005). In the case of a class  $c$  to be predicted based on training dataset  $D_{train}$  using  $n$  PointNets with initial value drawn from a normal distribution, the Bayesian model averaging provides final predictions based on the law of total probability:

$$p(c) = \sum_{a=1}^N p(c_b|m_a)p(m_a|D_{train}) \quad (3.5)$$

$$\hat{c} = \underset{c_b \in \mathcal{C}}{\operatorname{argmax}} \sum_{a=1}^N p(c_b|m_a)p(m_a|D_{train}) \quad (3.6)$$

As Equation 3.6 presents, the averaged model assigns higher weights to the candidate model which performs better for the specific class. The final prediction relies on the weighted average of the prediction scores.

### 3.6.3 Results

Five PointNet models were trained with different initial values. Figure 3.26 shows the CCR of each class from five different models.

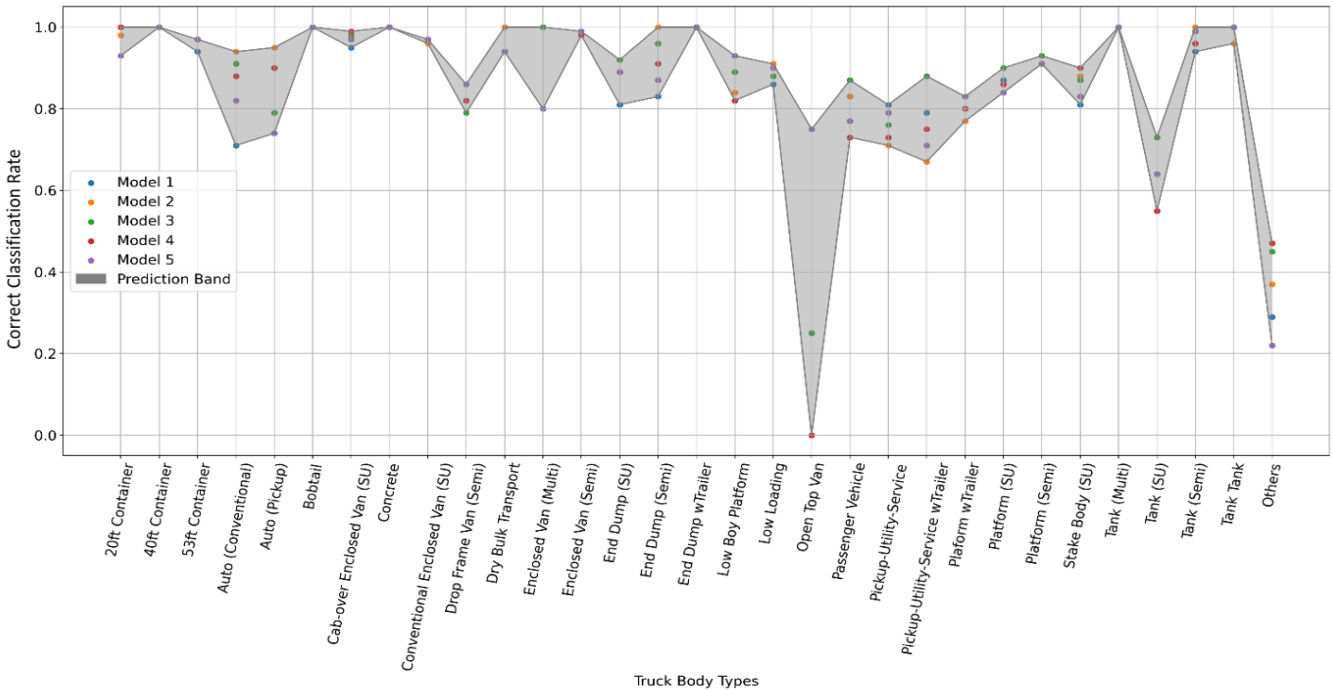


Figure 3.26 Prediction Variance Analysis

The same model structure yielded high variance to predict certain classes. For instance, model 4 obtained a CCR of 0 percent on predicting open-top vans, where model 5 was able to achieve a CCR of 75 percent on predicting the same class. Conversely, model 4 provided a CCR of 90 percent for single-unit stake body trucks. But the CCR for model 5 on predicting the same class was only 83 percent. Figure 3.26 thus reveals the need for an ensemble model. Auto (Conventional) and Auto (pickup) distinguish the tractor units of auto transport trucks. “Pickup-Utility-Service wTrailer” and “Platform wTrailer” are used to identify a straight driving unit pulling a small trailer. “End Dump wTrailer” considers an end dump truck pulling either a small trailer or another large dump trailer. “Other” represents

all the truck types that do not fit the definition of the previous 30 classes. “Tank Tank” represents a tank tractor with a tank trailer, which specifically belongs to Class 14 in the California-modified FHWA scheme (Quinley, 2010).

The results from SMA and BMA are presented in Table 3.4.

Table 3.4 Results from Test Dataset

	Model 1	Model 2	Model 3	Model 4	Model 5	SMA	BMA	Test Sample
20ft Container	0.98	0.98	1.00	1.00	0.93	1.00	1.00	59
40ft Container	1.00	1.00	1.00	1.00	1.00	0.99	0.99	196
53ft Container	0.94	0.97	0.97	0.97	0.97	0.96	0.96	170
Auto (Conventional)	0.71	0.94	0.91	0.88	0.82	0.91	0.91	34
Auto (Pickup)	0.90	0.95	0.79	0.90	0.74	0.89	0.89	19
Bobtail	1.00	1.00	1.00	1.00	1.00	1.00	1.00	109
Cab-over Enclosed Van (SU)	0.95	0.98	0.98	0.99	0.97	0.99	1.00	148
Concrete	1.00	1.00	1.00	1.00	1.00	1.00	1.00	16
Conventional Enclosed Van (SU)	0.97	0.96	0.97	0.97	0.97	0.97	0.97	362
Drop Frame Van (Semi)	0.82	0.82	0.79	0.82	0.86	0.82	0.79	28
Dry Bulk Transport	1.00	1.00	0.94	0.94	0.94	1.00	1.00	16
Enclosed Van (Multi)	0.80	0.80	1.00	0.80	0.80	0.80	0.80	5
Enclosed Van (Semi)	0.99	0.98	0.99	0.98	0.99	0.98	0.99	928
End Dump (SU)	0.81	0.89	0.92	0.89	0.89	0.88	0.88	26
End Dump (Semi)	0.83	1.00	0.96	0.91	0.87	0.96	0.96	23
End Dump wTrailer	1.00	1.00	1.00	1.00	1.00	1.00	1.00	7
Low Boy Platform	0.82	0.84	0.89	0.82	0.93	0.91	0.91	56
Low Loading	0.86	0.91	0.88	0.90	0.90	0.90	0.89	116
Open Top Van	0.00	0.25	0.25	0.00	0.75	0.25	0.00	4
Passenger Vehicle	0.83	0.83	0.87	0.73	0.77	0.80	0.80	30
Pickup-Utility-Service	0.81	0.71	0.76	0.73	0.79	0.79	0.79	94
Pickup-Utility-Service wTrailer	0.79	0.67	0.88	0.75	0.71	0.88	0.83	24
Platform wTrailer	0.80	0.77	0.80	0.80	0.83	0.83	0.83	30
Platform (SU)	0.87	0.86	0.90	0.86	0.84	0.90	0.89	135
Platform (Semi)	0.91	0.93	0.93	0.91	0.91	0.94	0.94	160
Stake Body (SU)	0.81	0.88	0.87	0.90	0.83	0.89	0.89	114
Tank (Multi)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	5
Tank (SU)	0.64	0.55	0.73	0.55	0.64	0.73	0.64	11
Tank (Semi)	0.94	1.00	0.99	0.96	0.99	0.99	0.99	83
Tank Tank	1.00	0.96	1.00	1.00	1.00	1.00	1.00	27
Others	0.29	0.37	0.45	0.47	0.22	0.37	0.18	49
Accuracy	0.92	0.93	0.94	0.93	0.93	0.94	0.94	3,084
Avg CCR	0.84	0.86	0.88	0.85	0.87	0.88	0.86	3,084

Note: SU: Single-Unit Truck; Semi: Tractors pulling Semi-Trailer; Multi: Tractors pulling multiple trailers. Cells labeled with red colors represent CCR lower than 0.80. Green colors highlight the benefits of using the SMA model.

After applying model averaging across five PointNet models, the number of classes with CCR value less than 80 percent was significantly reduced. The ensembled models

outperform most of the individual models in terms of accuracy, average class CCR, and F1 score. SMA and BMA presented the same level of accuracy according to these aggregated measurements. The SMA outperforms the BMA method in terms of model performance on drop frame vans (Semi), low loading truck, open-top vans, pickup/utility/service with trailer, and single-unit tank while the CCR of single-unit cab-over enclosed van was slightly reduced. The two-sided non-parametric Wilcoxon signed-rank test (Wilcoxon, 1946), was conducted to test if a significant difference existed between the results of SMA and BMA. The p-value of 0.02 showed the null hypothesis for the difference between SMA and BMA was significant, and at a significance level of 5 percent could be rejected. Therefore, the performance of SMA is significantly better than the BMA method.

A closer assessment of the minority classes found that the performance of the ensembled PointNet was not significantly biased towards the majority class since the minority class presented low variations in their body type design. Hence, the ensemble PointNet presents a promising result in solving truck body type classification problems.

For illustration purposes, in Table 3.5 and Table 3.6, the “Others” class was split into “Others (SU)” and “Others (wTrailer)” to denote single-unit trucks and trucks with trailer(s) that were misclassified with other types of trucks respectively.

Table 3.5 Confusion Matrix for single-unit truck and passenger vehicles (SMA Approach)

	Bobtail	Cab-over Enclosed Van (SU)	Concrete Mixer	Conv Enclosed Van (SU)	End Dump (SU)	Low Loading	Passenger Vehicle	Pickup-Utility-Service	Platform (SU)	Stake Body (SU)	Tank (SU)	Others (SU)	Test Counts	CCR
Bobtail	109	0	0	0	0	0	0	0	0	0	0	0	109	1.00
Cab-over Enclosed Van (SU)	0	147	0	0	0	0	0	0	0	1	0	0	148	0.99
Concrete	0	0	16	0	0	0	0	0	0	0	0	0	16	1.00
Conv Enclosed Van (SU)	1	2	0	351	0	7	0	1	0	0	0	0	362	0.97
End Dump (SU)	0	0	0	0	23	0	0	0	1	2	0	0	26	0.88
Low Loading	0	1	0	7	0	104	0	4	0	0	0	0	116	0.90
Passenger Vehicle	0	0	0	0	0	1	24	5	0	0	0	0	30	0.80
Pickup-Utility-Service*	0	2	0	2	0	1	3	74	6	4	0	1	94	0.90
Platform (SU)	1	0	0	2	0	0	0	3	121	5	1	2	135	0.79
Stake Body (SU)**	0	0	0	2	1	0	0	3	4	102	0	1	114	0.89
Tank (SU)	0	0	0	0	0	0	0	0	0	3	8	0	11	0.73
Other (SU)	0	0	0	0	0	1	0	3	4	1	0	18	27	0.67

Note: Cells labeled with red colors represent CCR lower than 0.80. The yellow cells highlight the correctly classified numbers. The grey cells point to the main causes of the misclassification. “Cov” is short for “Conventional”. \*This row is not added up to 94, since one of the pickup/utility/service trucks was misclassified with pickup/utility/service with a trailer. \*\* In this row, one stake body (SU) was misclassified to a semi-trailer platform. This was identified as mislabeling through visual verification.

Table 3.6 Confusion Matrix for a truck with Trailer(s) (SMA Approach)

	20ft Container	40ft Container	53ft Container	Auto (Conv)	Auto (Pickup)	Drop Frame (Semi)	Dry Bulk Transport	Enclosed Van (Multi)	Enclosed Van (Semi)	End Dump (Semi)	End Dump wTrailer	Low Boy Platform	Open Top Van	P/U/S wTrailer	Plaform wTrailer	Platform (Semi)	Tank (Multi)	Tank (Semi)	Tank Tank	Others (wTrailer)	Test Counts	CCR	(Sahin et al., 2020) CCR
20ft Container	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	1.00	0.96
40ft Container	0	195	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	196	0.99	0.98
53ft Container	0	0	164	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	170	0.96	-
Auto (Conv)	0	0	0	31	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	34	0.91	0.91
Auto (Pickup)	0	0	0	1	17	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	19	0.89	-
Drop Frame (Semi)	0	0	0	0	0	23	0	0	4	0	0	0	0	0	0	0	0	0	0	1	28	0.82	-
Dry Bulk Transport	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	16	1.00	-
Enclosed Van (Multi)	0	0	0	0	0	0	0	4	1	0	0	0	0	0	0	0	0	0	0	0	5	0.80	-
Enclosed Van (Semi)*	0	0	4	0	0	5	0	0	913	0	0	0	2	1	0	0	0	0	0	2	928	0.98	0.94
End Dump (Semi)	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	1	0	0	0	0	23	0.96	0.85
End Dump wTrailer	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	7	1.00	-
Low Boy Platform	0	0	0	0	0	0	0	0	0	0	0	51	0	1	0	4	0	0	0	0	56	0.91	-
Open Top Van	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	3	4	0.25	-
P/U/S wTrailer**	0	0	0	0	1	1	0	0	0	0	1	0	0	21	0	0	0	0	0	0	24	0.88	-
Plaform wTrailer	1	0	0	0	0	0	0	0	0	0	1	0	0	2	25	0	0	0	0	1	30	0.83	-
Platform (Semi)	0	0	4	1	0	0	0	0	0	1	0	1	0	0	0	150	0	0	0	3	160	0.94	0.94
Tank (Multi)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	5	1.00	-
Tank (Semi)	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	82	0	83	0.99	0.97
Tank Tank	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	0	27	1.00	-
Others (wTrailer)	0	0	7	0	1	1	0	2	4	0	0	0	2	0	1	2	1	1	0	18	40	0.45	-

Note: \* In this row, a semi-trailer enclosed van was misclassified to a pickup/utility/service truck. This was identified as mislabeling through visual verification.

\*\* "P/U/S wTrailer" represents the pickup/utility/service truck.

A comparison between the CCR values in Table 3.5 and Table 3.6 shows that the model was less competent in predicting single-unit trucks, where 18 percent of single-unit vehicles have an average CCR less than 0.80, while only 5 percent of trucks pulling trailer (s) have an average CCR less than 0.80. This was likely caused by the similarity across body types. For example, with different shapes of commodities or devices carried, single-unit platform trucks shared similar body configurations with pickup/utility/service trucks, single-unit stake body trucks, single-unit tank trucks, and single-unit dump trucks. In addition, the “passenger vehicle” class included 4-tire small pickups which shared a similar profile with 6-tire utility pickups that were categorized in the “pickup/utility/service” class.

Table 3.6 presents the confusion matrix of 19 truck body types, primarily including tractors pulling semi-trailers, tractors pulling a large single trailer, and tractors pulling multiple trailers. The body type confusion occurred primarily among auto transports, low boy platform, and semi-trailer platform trucks. Similar to the issues shown in single-unit trucks, the loading on the trailers is likely the cause of misclassifications across these three types. The performance of the SMA PointNet was compared with the state-of-art LiDAR-based classification model. SMA PointNet presented higher CCR values than the previous model across most classes, except semi-trailer platforms. In this sense, the ensemble PointNet can be considered superior to the state-of-the-art truck classification model (Sahin et al., 2020). Platforms loaded with 53ft box containers were misclassified as 53ft box containers loaded on a container chassis, which was not included in the previous trailer type classification scheme (Sahin et al., 2020). Classifying trucks in more detail naturally increases the chances of misclassification among similar body types. Therefore, balancing



the total number of trucks that can be classified and high CCR values across all classes is critical.

### **3.6.4 Conclusion**

To fill the truck monitoring gaps on rural highway corridors, this chapter presented novel LiDAR-based truck classification methods through the development of a new truck point cloud reconstruction framework that was able to retain a wide LDZ and accurately classify trucks on the basis of the FHWA-CA scheme and detailed truck body configurations. The data used for modeling was collected from a horizontally-oriented multi-array 3D LiDAR sensor, which had the ability to capture a wide field of view of the roadway. In this case, even though vehicles traveling in the outermost lanes presented in front of the LiDAR sensor for a short period of time and occluded vehicles traveling in the corresponding inner lanes, the point cloud originating from those occluded vehicles could be retrieved from consecutive frames. The sparse point clouds from individual frames resulting from a low vertical resolution were enriched by aggregating multiple frames associated with the same truck. Subsequently, the lower profile of the reconstructed vehicle point cloud was extracted and used as inputs for the deep neural network to classify vehicles based on the FHWA classification scheme. The classification model with the reconstruction framework outperforms the state-of-the-art axle-based classification model using LiDAR sensors in terms of both their accuracy and robustness. This LiDAR-based FHWA model presents a 79 percent average CCR. Furthermore, the proposed model is capable of accurately distinguishing Classes 3, 5, and 8 which have overlapping axle configurations with a 98 percent and an 86 percent correct classification rate, respectively. Further, Classes 8 and 9 can be classified correctly with 84 percent and 99 percent CCR even though they share very similar body configurations.

In order to further classify trucks in their detailed body configurations, this chapter adopted a deep representation learning algorithm, PointNet. PointNet successfully learned the basic characteristics of each truck class by selecting the critical features from each preprocessed point cloud. Finally, two model ensemble strategies, SMA and BMA, were explored to improve the generality of the model and to further enhance the model performance. The LiDAR-based truck body type classification model was able to classify heavy-duty trucks in much more detail, with a close relationship to their industry affiliations. For example, the new model could accurately distinguish low boy platforms from general flatbed trucks, where these two types of platform trucks are designed to carry different types of payloads. This model was able to classify 31 different vehicle types (advantageously mainly trucks) and achieve an average class CCR of 90 percent for both a truck with trailer (s) and single-unit vehicles. Remarkably, the proposed method was able to distinguish 53ft containers and semi-trailer enclosed vans with over 95 percent CCR even though they share very similar physical characteristics, which is a significant improvement over previous models using the integration of WIM and inductive signature data (Hernandez et al., 2016), as well as LiDAR (Sahin et al., 2020).

In the future, more LiDAR data will be collected from other detection sites to test the transferability of the proposed model. Furthermore, multi-lane truck classification applications can be explored as the horizontal orientation of the LiDAR permits capturing a full 360-degree field of view.

## **CHAPTER 4 : A Self-Learning Framework for Truck Classification System through the Integration of Advanced Inductive Loops and Multi-array LiDAR Sensor**

### **4.1 Introduction**

As mentioned in Chapter 2, inductive loop sensors are widely deployed across the U.S. Each detection site comprises inductive loop sensors that monitor traffic traversing the detection site across all lanes. The vehicle signature data retrieved from the inductive loop sensors have been adopted to develop vehicle classification models to classify trucks on the basis of the FHWA-CA (Chapter 2) and the truck body configuration scheme (Hernandez, 2014). Such detection systems allow us to monitor truck activities extensively on major freeways. However, the performance of these models is expected to degrade as truck fleets turnover and newer truck models and configurations begin to operate on highway networks in the future. The newer trucks with different body-style designations may generate distinct signature patterns which may not be accurately classified by a legacy signature-based classification model since the signature-based model has been proved that primarily infers FHWA classes through their body configurations. On the contrary, the roadside LiDAR sensor is capable of capturing the detailed truck profile by merging multiple consecutive frames of the same truck and identifying their FHWA classes by recognizing their axle and general body types with a relatively high level of accuracy. By utilizing the superiorities of these two complementary data sources – the inductive signature data and the LiDAR vehicle point cloud data – a self-learning framework was designed to alleviate the obsolescence of the truck classification system, as well as to enhance the system resilience for its long-term operations.

In this chapter, both qualitative and quantitative analyses were performed on the input features for the signature- and LiDAR-based vehicle classification models to identify the key features that were used to distinguish trucks on the basis of the FHWA classification scheme. In this self-learning framework, the performance of the legacy signature-based model was initially validated at the detection sites where the inductive loop detector and the LiDAR were paired. At those detection sites, the LiDAR-based classification model served as an automated data labeling platform to generate labels for the newly collected signature data. The data labeled by LiDAR sensors were used for model validation. If the performance of the legacy signature-based classification model degrades and meets a given model-updating criterion, the legacy signature-based FHWA classification model will be calibrated and updated with the newly LiDAR-labeled data to the state-of-the-art model using an adaptive transfer learning framework which exploit both the archived data and the newly labeled data from the LiDAR sensor to enhance the model performance. Finally, the legacy signature-based classification model will be updated across all loop detection sites to produce reliable FHWA vehicle classification counts on both spatial and temporal levels. The integration of the inductive loop detectors and the LiDAR sensors enables the self-learning ability of the truck classification system without a labor-intensive manual data annotation process. The adaptive transfer learning module in this framework balanced the model performance on the statewide and the state-of-the-art datasets to remedy the model overfitting issue on each dataset. Meanwhile, this module reduces the computation burden of the model calibration procedure.

## **4.2 Interpreting DNN-based Vehicle Classification Model Predictions**

From a theoretical perspective, the multiple layer structure with a non-linear activation of the deep neural network (DNN) enables it to approximate any complex mapping function (Goodfellow et al., 2016). From a practical perspective, the DNN model has been adopted to solve various vehicle classification problems (Chapter 2 and 3) with relatively high accuracy in the transportation domain. However, the transportation literature generally overlooks the importance of interpreting feature characteristics that were used by DNN models, which is essential to facilitate the sensor integration task which is the focus of this section. Therefore, this section explains how the features obtained from different sensors were used by the DNN models to make predictions. First, this section provided a qualitative analysis of the input features extracted from inductive loop signatures and LiDAR point clouds. The interpretation of the input features is primarily based on human intuition. Second, the SHapley Additive exPlanations framework was used to quantify the importance of features for making certain predictions.

### **4.2.1 Human Intuition**

The initial intention of creating neural network models was to let the model mimic the functionality of human brains, where a human recognizes certain objects according to their distinct features. Hence, the input features of the neural network model are required to contain essential features that differentiate predefined classes. The FHWA vehicle classification scheme was defined based on vehicles' axle and general body configurations. Therefore, the feature vectors that are used by the FHWA classification models are expected

to carry vehicle axle configurations. Figure 4.1 shows the reconstructed point cloud of a 40ft container truck and its corresponding vehicle signature.

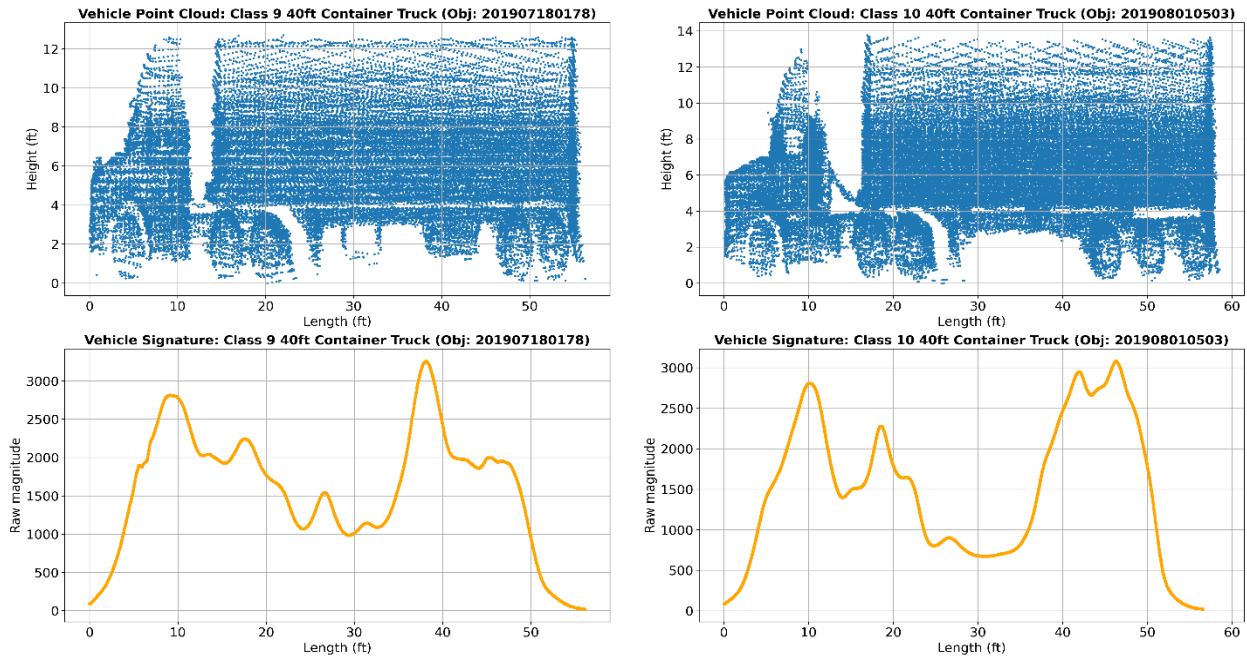


Figure 4.1 A Vehicle Point Cloud and A Signature from the Same 40ft Container

In Figure 4.1, there is no visually identifiable correspondence between the vehicle signature and its axle location. On the other hand, the reconstructed vehicle point clouds present a clear contour of the undercarriage of the truck. Therefore, from the perspective of human intuition, vehicle signature data contains limited information in terms of vehicle axle configuration while the vehicle lower profile of the LiDAR point cloud data reveals the essential information for the task of FHWA vehicle classification.

## 4.2.2 Interpreting Vehicle Classification Model Predictions

### 4.2.2.1 The Model Interpretation Method

A qualitative analysis of input features for signature- and LiDAR-based model provides insights as to how each model presumably make predictions. However, the importance of

the input features for the classification model is yet to be quantified. Lundberg & Lee (2017) proposed a unified framework, SHapley Additive exPlanations (SHAP), for interpreting complex machine learning model predictions. SHAP adopts Shapley value from coalitional game theory to interpret the prediction result of an instance, where the Shapley value presents the importance of each feature in the instance of the final prediction. SHAP treats each feature value as the player in the cooperative game. SHAP calculates the marginal contribution of a feature (players) while making the prediction (running the game) with or without it under all the possible subset of features (players) in the instance.

#### **4.2.2.2 Interpreting Vehicle Classification Model using SHAP**

SHAP feature importance was utilized in this session to identify the key features that are used for the signature- and LiDAR-based classification model, respectively. The features with higher absolute Shapley values indicated that they provide a more essential contribution to make a certain prediction. This research focuses on the global importance of each feature within each FHWA class. Therefore, the sum of absolute Shapley values per feature across

the instances in every class was calculated. Figure 4.2 shows samples of instances that were used in the signature-based FHWA classification model.

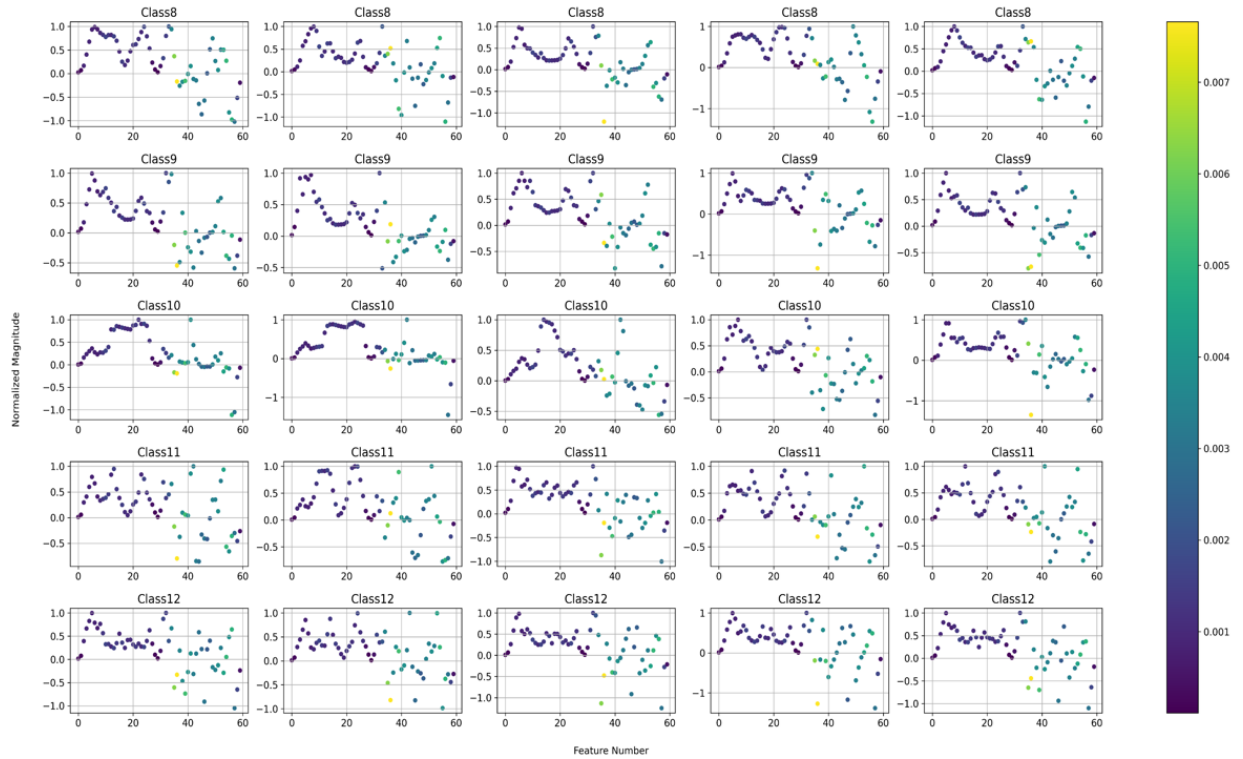


Figure 4.2 SHAP Values for Features from Signature Data

The sum of absolute Shapley values for each feature is denoted by colors. According to the color bar shown in Figure 4.2, the lighter the color presents, the greater the Shapely value is. For the signature-based model, the higher Shapely values are distributed mainly on the features which represent the magnitude differences of the signatures and spread all over the truck body. This implies that the signature-based FHWA classification model makes predictions by looking at the overall truck body instead of focusing on their axle configuration. However, the only difference among Classes 8, 9, and 10 (Similarly for Classes



11 and 12) is the number of axles. Therefore, it is hard for the signature-based model to differentiate those classes with the same body type but different axle configurations.

On the contrary, the LiDAR-based FHWA classification focuses on the features located around the truck axle (Figure 4.3).

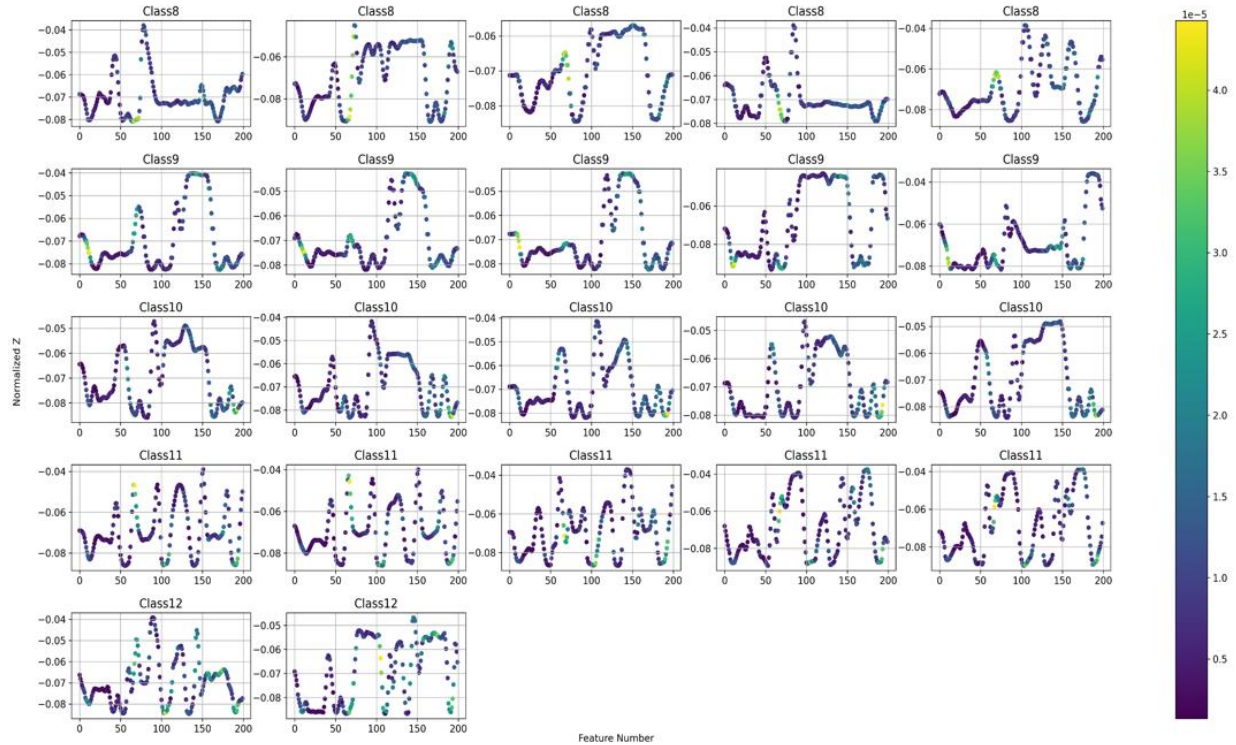


Figure 4.3 SHAP Values for Features from Point Cloud Data

For example, as Figure 4.3 shows, the features with higher Shapely values to determine Classes 8, 9, and 10 are positioned at the drive axle on the tractor unit and the tandem axle on the trailer unit, where those axle configurations are key to distinguishing their classes based on the FHWA definition (Figure 4.4).

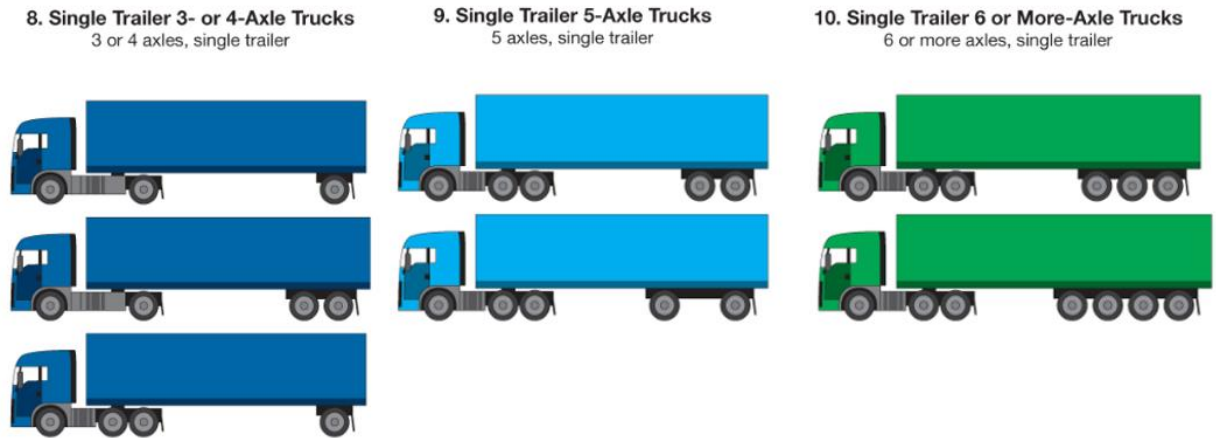


Figure 4.4 Differences among Classes 8, 9, and 10

Similarly, the essential features that are used by the LiDAR model to determine Classes 11 and 12 are located at split tandem axles on the trailer units (Definition finds in Figure 4.5).

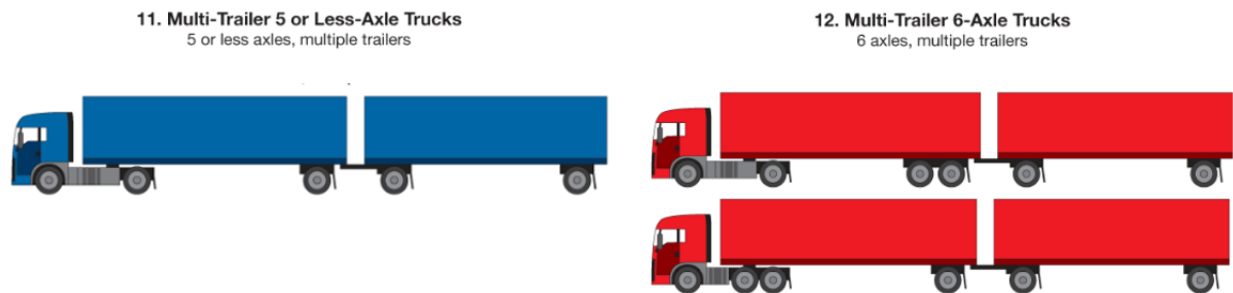


Figure 4.5 Differences between Classes 11 and 12

#### 4.2.2.3 Summary

In summary, both the qualitative and quantitative analysis shows that the signature-based classification model infers the FHWA classes according to the truck body types while the LiDAR-based classification model recognizes the truck classes based on the definition of the FHWA classification scheme. As the truck body configuration changes over time or across states, the performance of the signature-based model may degrade. However, the

performance of the LiDAR-based model is expected to retain the same if the definition of the FHWA classification scheme is not changed.

### **4.3 A Self-learning Framework for Truck Monitoring System**

The advanced loop sensors are widely deployed across the U.S. The vehicle signature data retrieved from such sensors provide a promising FHWA classification counts data on major freeways. However, as mentioned in the previous section, the signature-based model is possibly obsolete. Conversely, the LiDAR sensors are limited in their spatial coverage. But its performance may not degrade along the planning horizon since it is capable of identifying FHWA classes based on its definition. This section introduced a self-learning framework for the truck classification system by utilizing the advantages of inductive loop signature data and 3D point cloud data from the LiDAR sensor to reduce system obsolescence and enhance its resilience.

#### **4.3.1 The Self-learning Framework**

The self-learning framework proposed in this dissertation contains three major modules: the LiDAR labeling module, the model validation module, and the adaptive transfer learning module. The self-learning truck classification system is presented in Figure 4.6.

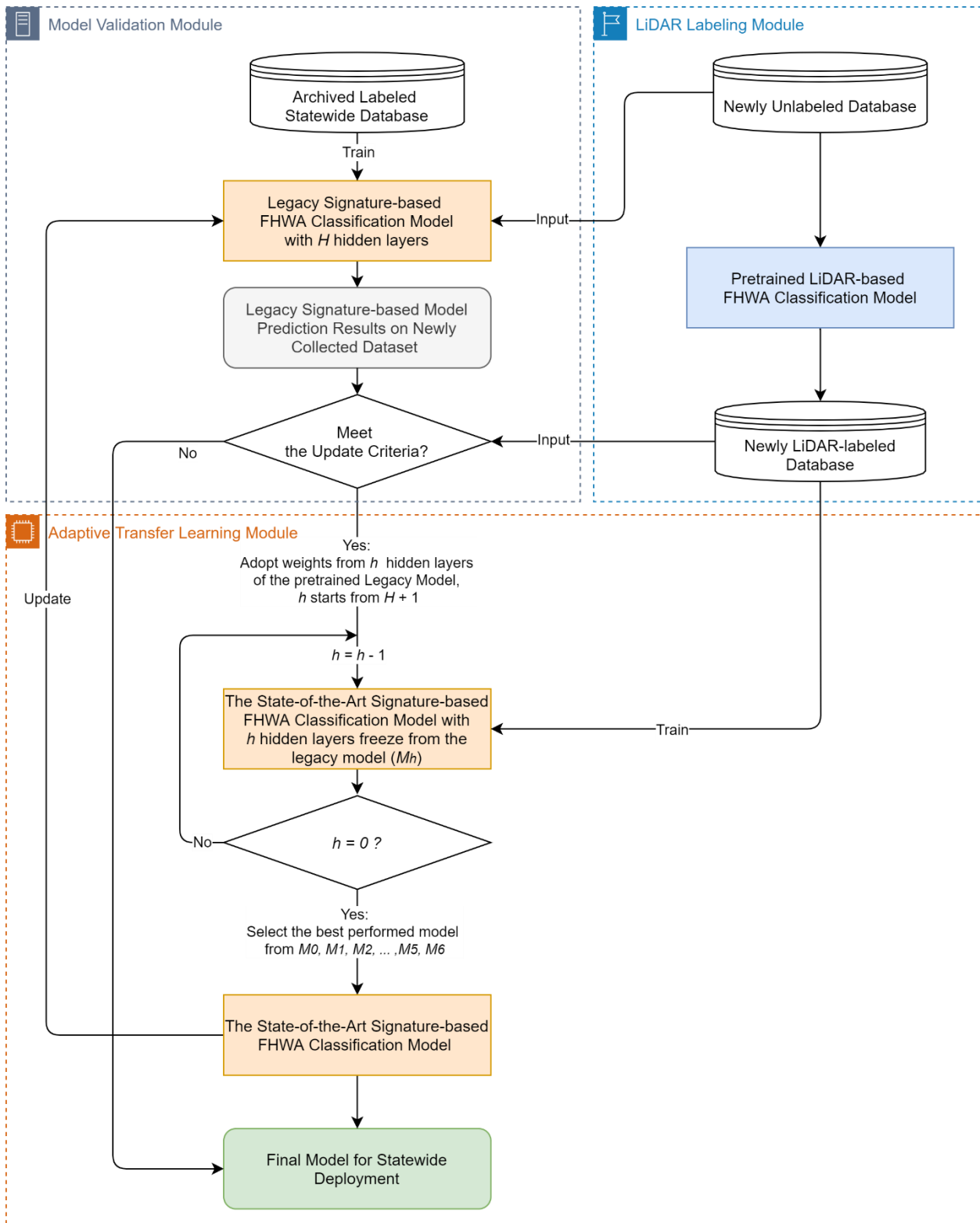


Figure 4.6 Self-Learning Truck Classification System

#### **4.3.1.1 The LiDAR Labeling Module and the Model Validation Module**

The designed self-learning framework includes two types of detection sites: the inductive loop sensor only site and the sensors integration site, where the inductive loop sensors and LiDAR sensors are paired. The newly collected data from the sensor integration site is labeled by a pre-trained LiDAR-based FHWA classification model in the LiDAR labeling module of the self-learning framework and the legacy signature-based classification model is validated via the LiDAR-labeled data in the model validation module of the framework. If the performance of the legacy signature-based FHWA classification model on one of the classes significantly degraded and consequently met the model-upgrade criteria, the legacy signature-based model will be calibrated using an adaptive transfer learning framework.

#### **4.3.1.4 The Adaptive Transfer Learning Module**

The basic idea of transfer learning is to utilize the knowledge which has been learned from one task (Task 1) to improve generalization in a different task (Task 2) (Goodfellow et al., 2016). Under the context of the truck classification system, the legacy signature-based FHWA model trained on the large archived dataset was designed to recognize the state-of-the-art truck designations. However, as the truck fleets turn over and newer truck body types came to the market, the previous “state-of-the-art” truck model will be obsolete. Thus, the legacy model may not be able to accomplish the new task—recognizing the newer truck models. The traditional way that is used to address such an issue is to manually label the new dataset and retrain the signature-based model with the overall dataset, including the archived and the new label signature data. Nevertheless, such an approach becomes computationally inefficient since the model needs to be trained with an incremental dataset at each time. Therefore, this research adopted the transfer learning framework to improve the

computation efficiency as well as the performance of the self-learning framework. Figure 4.7 shows how the transfer learning framework is applied to the truck classification problem.

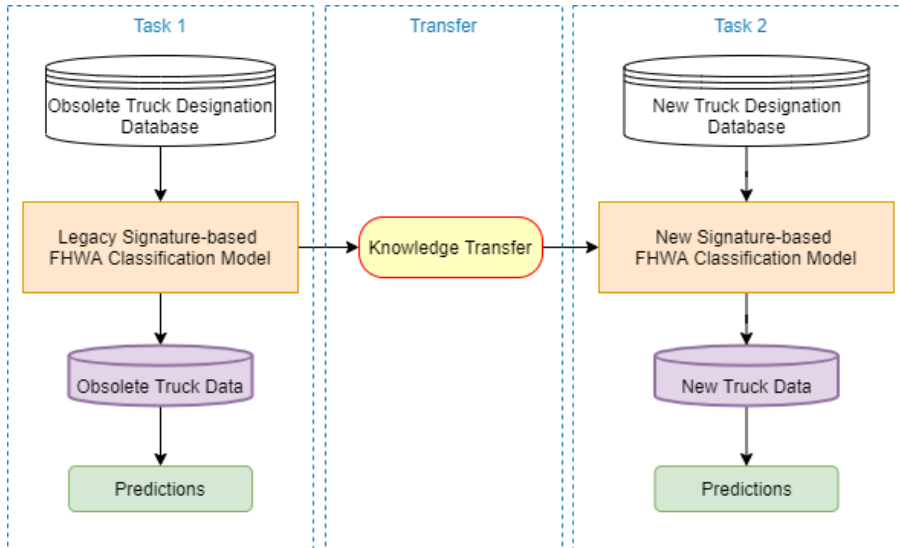


Figure 4.7 Illustration of Transfer Learning

Task 1 refers to predicting truck types with obsolete truck designations while task 2 indicates the prediction of truck types with newer body designations. Instead of retraining the whole model with an incremental dataset, the transfer learning framework transmits the knowledge from task 1 to task 2 by freezing several hidden layers of the neural network model used in task 1 and retrains the last few layers with the newly labeled dataset (new truck designation dataset).

Figure 4.8 presents the legacy signature-based FHWA classification with different numbers of layers fixed.

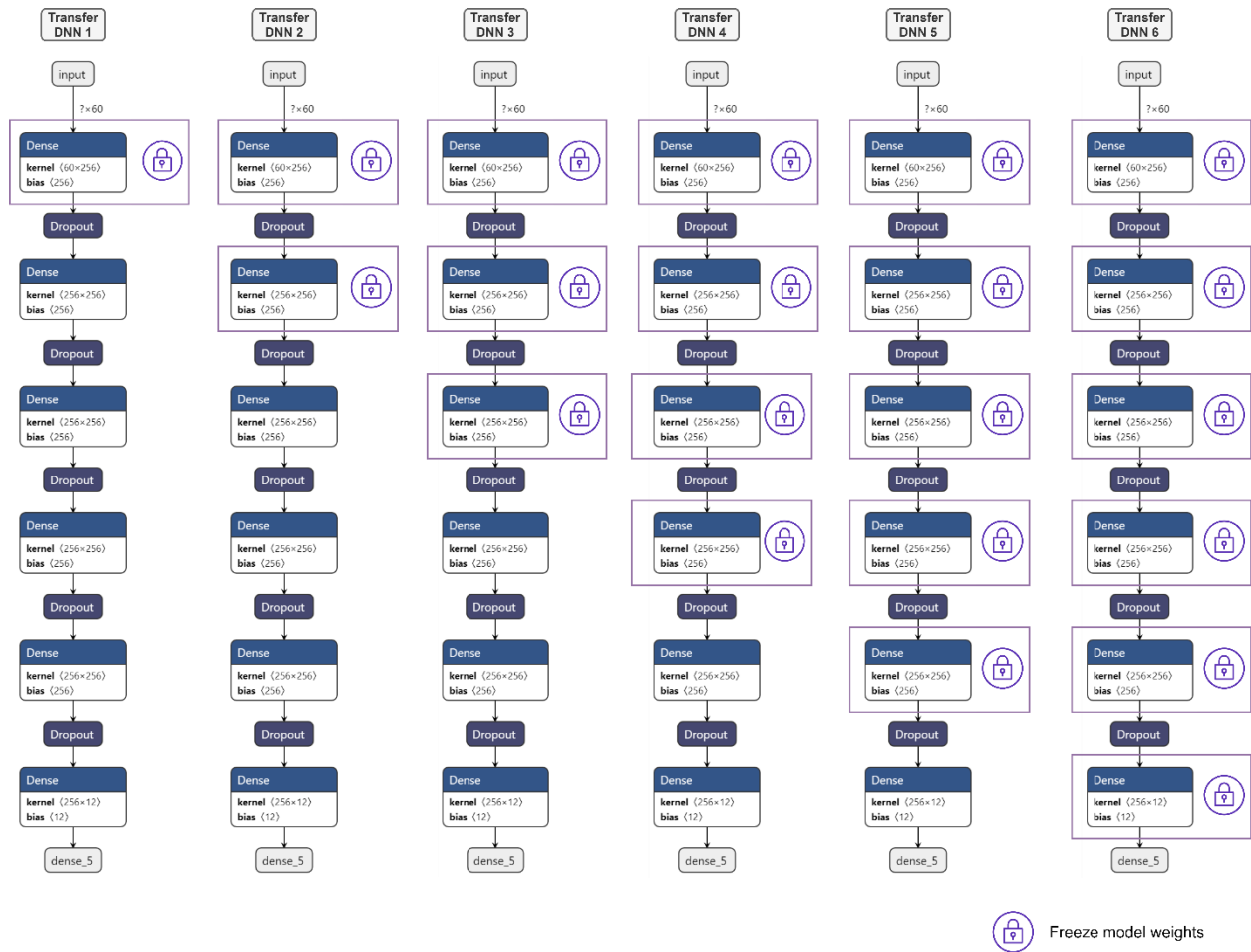


Figure 4.8 Adaptive Transfer Learning for Truck Classification

For example, transfer DNN 1 represents a transfer learning model structure that fixes the weights of the first layer of the neural network model and fine-tunes the rest of the layers with the newly labeled data. Transfer DNN 6 shows the model structure which adapts the weights entirely while retraining the model with the new data. It can also be considered as directly using the legacy signature-based model on the newly labeled dataset. In this study, 6 different transfer learning models (Transfer DNN 1 to 6) were designed and trained. Different from the conventional transfer learning framework which only focuses on improving the performance of the new task, this research developed an adaptive transfer learning framework to serve the needs of transportation applications. Since the newly

collected data can only be labeled at the sensor integration site, the model design which restrictive to the sensor integration site may compromise its performance on those statewide deployed loops only sites. Therefore, the designed adaptive transfer learning framework should be able to balance the model performance between the newly labeled dataset and the state-wide dataset. In the adaptive transfer learning framework, the best-performed model is determined by the weighted average of the model performance on the statewide dataset and the newly collected dataset.

## 4.3.2 System Implementation

### 4.3.2.1 Data Description

The data used for the implementation of the designed self-learning framework comprises two parts, the statewide and the state-of-the-art dataset.

- The statewide dataset ( $D_{sw}$ ): The statewide dataset is the dataset that was used to develop the signature-based vehicle classification model in Chapter 2; the spatial distribution of the detection sites are presented in Figure 2.17. The signature data in this dataset were collected at 20 different detection sites equipped with either 6ft square or round loop sensors across California in 2012, 2013, and 2016. In this chapter, it is also called the “Archived dataset”. A total of 34,039 vehicle signature records in the statewide dataset were used for the framework implementation in this chapter. Seventy percent of the data from the statewide dataset ( $D_{sw}^{train}$ ) was used to train the legacy signature-based model and the rest of the data ( $D_{sw}^{test}$ ) was used for the validation purpose.



- The state-of-the-art dataset ( $D_n$ ): The state-of-the-art dataset refers to the data collected from a sensor integration site located at the off-ramp area at San Onofre truck scale in Southern California. The data were collected from July 18<sup>th</sup> to August 5<sup>th</sup> in 2019. Overall, 10,024 signatures were recorded. Seventy percent of them were used to train the state-of-the-art signature-based FHWA classification model. The class labels used in the training set ( $D_{nl}^{train}$ ) were produced by a pre-trained LiDAR-based FHWA classification model while the signatures in the test set were labeled by both LiDAR-based model and the ground truth via visual verifications. The LiDAR-labeled testing set was denoted by  $D_{nl}^{test}$  and the ground truth-labeled testing set was represented by  $D_{ng}^{test}$ .

The datasets that are used for system implementation are illustrated in Figure 4.9.

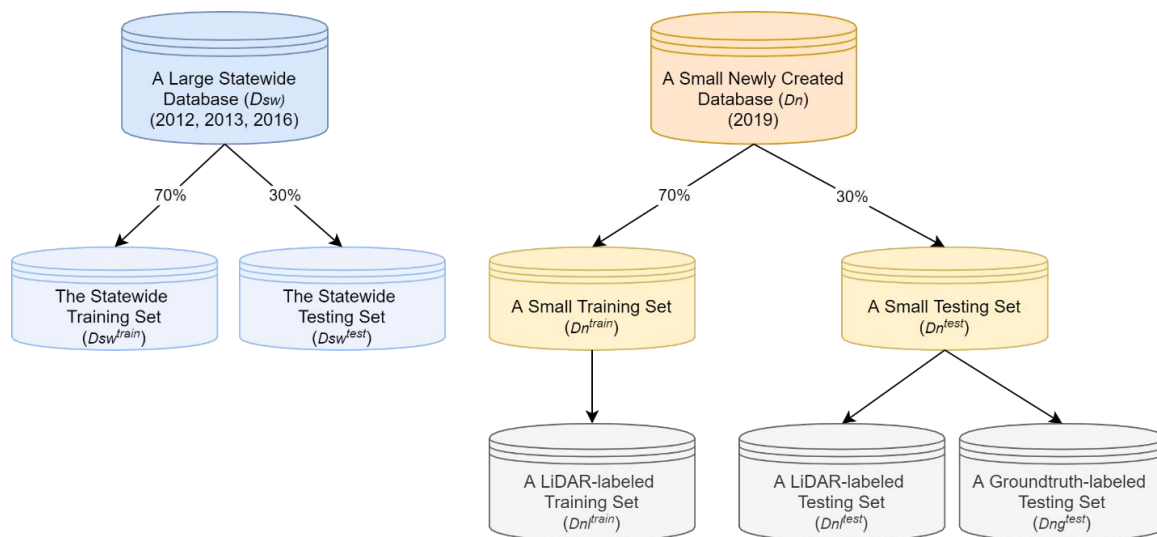


Figure 4.9 Data Description

Since the Class 13 trucks were not observed in  $D_n$  and Classes 2 and 3 were a lack of training samples in  $D_n$ , Class 13 was excluded in  $D_{sw}$  and Classes 2 and 3 were combined in  $D_{sw}$  to make the class numbers consistent between  $D_n$  and  $D_{sw}$ .

#### 4.3.2.2 Implementation

First, the legacy signature-based model was trained on  $D_{sw}^{train}$  and tested on  $D_{sw}^{test}$ . Then, the model was validated on  $D_{nl}^{test}$  and the performance degradation class was identified according to the reduction of the CCR value for every class. Subsequent, the overall  $D_n$  was undersampled based on the sample size of the degradation class. Conventionally, the undersampling method that is used to address imbalanced dataset issues in predictive models could result in the overfitting of a certain class. However, the undersampling method perfectly suited the target of the self-learning framework. During the model validation process, the consistency of the model performance on a certain class indicates that there is a limited number of new truck designations observed in this class. Thus, introducing more training samples will not provide a significant contribution to improving the CCR value of this class but let the model skew to it if it is a majority class. On the contrary, the performance degradation for certain classes could be caused by the truck type variations for this class in the new dataset. Therefore, all the samples should be considered during the training process to allow the model to recognize the new truck designations. The undersampling procedure is performed as follows. The performance degradation class is set as a target class. If the sample size of a certain class is larger than the sample size of the target class, this class will be randomly downsampled to align with the sample size of the target class. If the sample size

for a class is less than or equal to the sample size of the target class, all the samples belonging to this class will be retained for both training and testing purposes.

Finally, the adaptive transfer learning framework was applied. The pre-trained legacy signature-based model was retrained with various number of hidden layers fixed, where the undersampled  $D_{nl}^{train}$  was taken as the training set. Overall, 6 adaptive transfer learning models were trained and tested on both the  $D_{sw}^{test}$  and  $D_{nl}^{test}$ .

### 4.3.2.3 Result Analysis

The boxplot in Figure 4.10 presents the results of 30 different runs of the legacy signature-based classification model. The yellow line highlights the lower outlier boundary of the CCR of the 30-runs results on the statewide dataset.

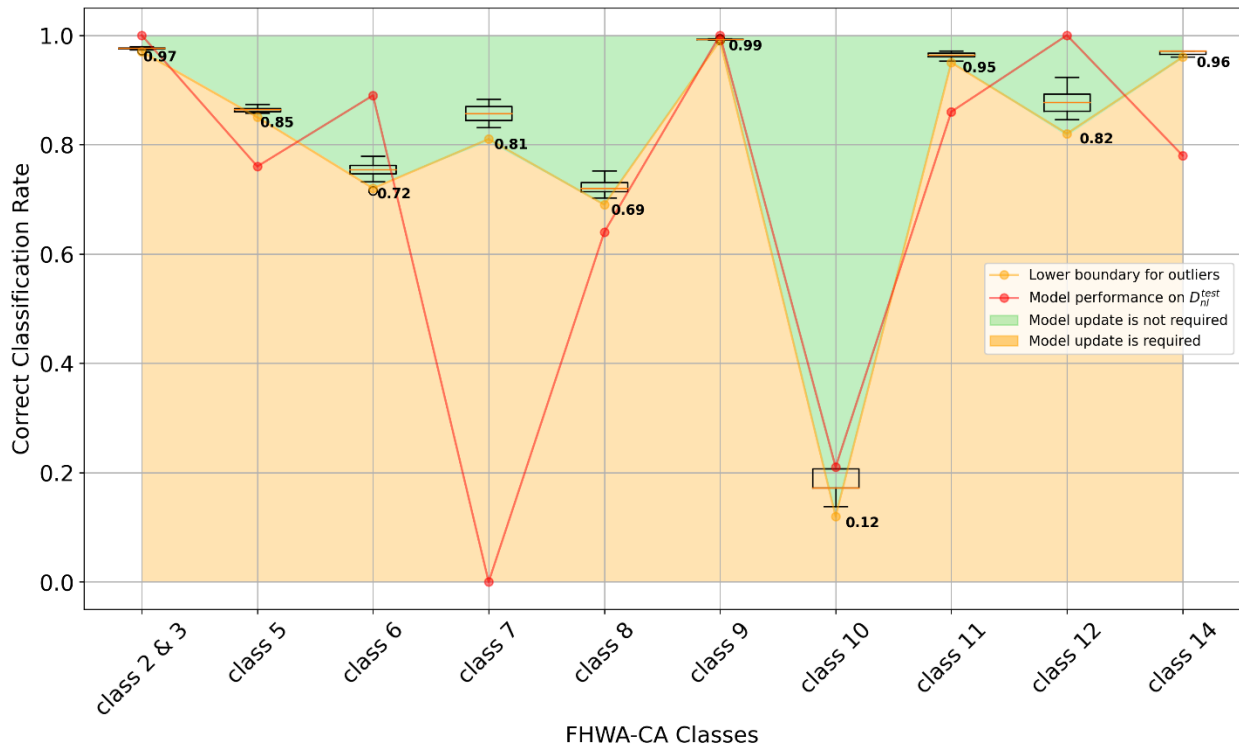


Figure 4.10 Model Validation

The red line in Figure 4.10 presents the model performance on the newly collected dataset. According to Figure 4.10, the CCR of Classes 5, 7, 11, and 14 on  $D_{nl}^{test}$  lie below the CCR lower outlier boundary on the  $D_{sw}^{test}$ , especially for Class 7. Therefore, Class 7 is identified as the performance degradation class on which the self-learning framework is going to focus.

After the model validation process, the adaptive transfer learning module was applied. Figure 4.11 presents the model results of the transfer learning model with a different number of layers fixed on both the statewide dataset and the newly collected dataset.

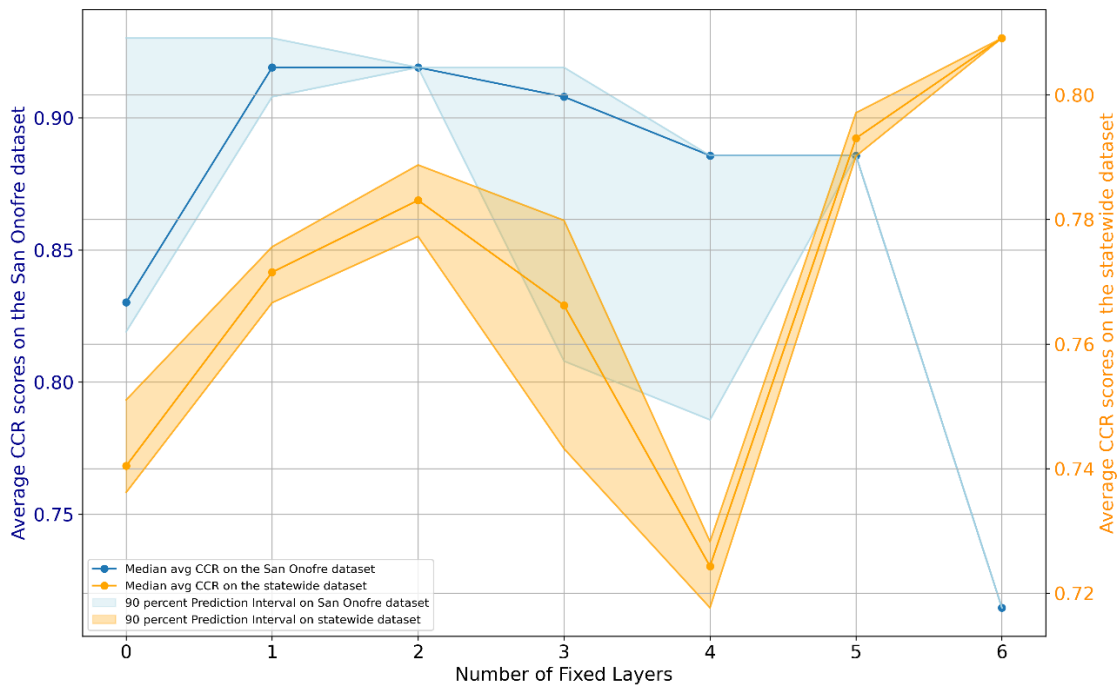


Figure 4.11 The Testing Results of the Adaptive Transfer Learning on the San Onofre dataset and the Statewide dataset

The blue line shows the model performance on the newly collected dataset. Fixing 0 hidden layer means that the legacy signature-based model was completely retrained using the small

dataset collected from the San Onofre detection site. With 1 hidden layer fixed, the model adopted the knowledge transferred from the previous task (predicting truck classes on the statewide dataset) and improved the model performance on the new task (predicting truck classes on the newly collected dataset). However, with the number of fixed layers increases, the model starts to overfit the statewide dataset. Thus, as the number of fixed layers increases, the model performance reduces on the San Onofre dataset while it rises on the statewide dataset. Besides, the prediction interval becomes much narrower when more hidden layers are fixed. The transfer learning framework also reduced the prediction variance and improved the generality of the model. Interestingly, the yellow graph shows a dip when the number of fixed layers equal to 4. Potentially, this could imply that fixing four hidden layers of neural network model may not serve as a good feature extractor for the legacy signature-based classification model. In this research, the statewide dataset was assumed to be two times more important than the newly collected data. Therefore, according to the weighted average of the median CCR on both the statewide and newly collected dataset, the model with 5 fixed hidden layers was selected as the model that performed best.

Table 4.1 presents the results of 3 different models: the model that was only trained using the newly collected dataset (Fix 0 layers), the standalone legacy signature-based model (Fix all layers), the best performed adaptive transfer learning model with 5 hidden layers fixed.

Table 4.1 Model Comparison

	Fix 0 hidden Layers		Fix all hidden Layers		Fix 5 hidden layers	
Detection Sites	CCR on $D_{sw}^{test}$	CCR on $D_{nl}^{test}$	CCR on $D_{sw}^{test}$	CCR on $D_{nl}^{test}$	CCR on $D_{sw}^{test}$	CCR on $D_{nl}^{test}$
Class 2	0.90	1.00	0.98	0.78	0.94	0.89
Class 5	0.72	0.78	0.86	0.78	0.63	0.78
Class 6	0.65	0.89	0.74	0.89	0.81	0.89
Class 7	0.68	1.00	0.86	0.00	0.68	0.89
Class 8	0.68	0.89	0.73	0.67	0.61	0.78
Class 9	0.92	1.00	0.99	1.00	0.95	0.89
Class 10	0.45	1.00	0.17	0.40	0.52	1.00
Class 11	0.79	0.86	0.96	0.86	0.95	0.86
Class 12	0.68	0.00	0.83	1.00	0.91	1.00
Class 14	0.97	0.89	0.97	0.78	0.98	0.89
Avg CCR	0.74	0.83	0.81	0.71	0.80	0.89
Weighted* Avg CCR		0.77		0.78		0.83
Approximate Run Time (min)		05:00		0:08		00:10

Note: \*Here, the statewide dataset has been considered two times more important than the newly collected dataset at San Onofre.

As Table 4.1 shows, with 0 hidden layers fixed, the model tends to overfit the new dataset and has poor performance on the statewide dataset. On the contrary, with all hidden layers fixed, the model is not capable of correctly classify newer trucks in Class 7. The CCR and F1 score becomes 0 for Class 7 trucks on the newly collected dataset. The adaptive transfer learning framework balanced the performance on both the statewide dataset and the newly collected dataset without compromising the computation efficiency. This framework improves the overall model performance in terms of the weighted average CCR value and alleviated the model obsolescence issue in an automated fashion.

The best-performing model has been evaluated on both the ground truth-labeled dataset and the LiDAR-labeled dataset. Table 4.2 presents model validation results on the ground truth-labeled dataset as well as the LiDAR-labeled dataset. The model presents a similar prediction performance in terms of CCR values on all classes except Classes 2, 3, and

5. The prediction results evaluated by the LiDAR-labeled dataset underestimated the true performance of the best model regarding CCR value. This could be likely caused by the poor performance of the LiDAR-based FHWA model on Classes 2 and 3 vehicles (Table 3.3). For the model performance on Class 5 vehicles, the evaluation results are high on the LiDAR-labeled test set while low on the ground truth labeled test set. In this case, the LiDAR-based and signature-based models might be biased towards the same directions.

Table 4.2 Model Validation on Ground truth-labeled dataset vs. LiDAR-labeled dataset

	The Best model CCR on Ground truth-Labeled test set	The Best model CCR on LiDAR-Labeled test set
<b>Class 2 &amp; 3</b>	1.00	0.89
<b>Class 5</b>	0.67	0.78
<b>Class 6</b>	0.89	0.89
<b>Class 7</b>	0.89	0.89
<b>Class 8</b>	0.78	0.78
<b>Class 9</b>	0.89	0.89
<b>Class 10</b>	1.00	1.00
<b>Class 11</b>	0.86	0.86
<b>Class 12</b>	1.00	1.00
<b>Class 14</b>	0.89	0.89
<b>Avg over All Classes</b>	0.89	0.89

The performance of LiDAR-based classification is sufficient to serve as a data labeling platform for the self-learning framework, especially for heavy-duty trucks. The LiDAR-based model still needs to be further enhanced on predicting pickup trucks and passenger vehicles, in order to provide reliable labels for the validation and calibration of the signature-based model across all classes.

#### 4.4 Conclusion and Discussion

This chapter explored the use of advanced models using roadside LiDAR sensors, which have been proved to be capable of directly capturing truck axle configuration to update a legacy inductive signature model using a self-learning framework. In this concept, LiDAR sensors

can be installed at a strategic subset of existing inductive loop signature sites to capture the characteristics of new trucks, and serve as an automated data labeling platform for enhancing the legacy signature-based classification model with its ability to directly capture axle configuration of trucks using dense reconstructed 3D point-clouds. The legacy model contains information that is needed to characterize a significant proportion of the truck population still in operations. However, a full replacement of the existing legacy signature-based model to enable its ability to recognize new trucks requires a cost-intensive periodic data collection, annotation, and model training process. Therefore, this chapter developed a transfer learning framework to enhance the generality of the legacy signature-based model without a costly full model migration. In the transfer learning framework, the legacy signature-based model was updated with a small LiDAR-labeled newly collected dataset by fixing different numbers of hidden layers of the DNN model. Then, the transfer DNN models with different fixed layers were analyzed on both the archived signature and the newly collected dataset. The model with the optimal number of fixed layers was selected according to the weighted average of the model performance on both datasets such that the signature-based transfer DNN model is capable of correctly classifying both old and new truck designations. The integration of the inductive loop detectors and LiDAR sensors enables the self-learning ability of the truck activity monitoring system without a labor-intensive data annotation process. This framework enhances the resilience of the signature-based FHWA classification model and reduces the obsolescence issue of the existing truck classification system.

This self-learning framework can be potentially adopted to automatically upgrade and calibrate the signature-based FHWA classification when it is deployed outside California.



For example, triple-trailer trucks in Class 13 are permitted in many other states. But they are restricted in California and not included in our training dataset. When the model is re-deployed in a state, where triple-trailer trucks are permitted. The designed self-learning framework is expected to be able to learn such information and further enhance its performance in predicting Class 13 vehicles.

## **CHAPTER 5 : Conclusion, Discussions, and Future Works**

### **5.1 Conclusion**

This dissertation designed a resilient spatiotemporal truck monitoring framework using advanced loop sensors and LiDAR sensing technologies. The major contributions of this dissertation can be summarized in three parts:

1. For major highway truck monitoring using inductive loop sensors, the model bias issue that occurred in the literature was confirmed. The previous signature-based truck speed estimation and FHWA classification model overlooked the model performance on minority classes, even though many of these minority classes may pose a disproportionately adverse impact on the pavement structure and roadway environment. In this dissertation, a signature-based individual truck speed estimation model and an FHWA vehicle classification model were developed with an emphasis on the enhancement of the model performance concerning minority classes without compromising the model performance on majority classes. The two models developed in this dissertation are capable of providing reliable truck speed and FHWA class data on major freeways.
2. In order to address the truck surveillance gap on rural highways for long-term operations, a non-intrusive sensor solution – Multi-array LiDAR sensor – was used to develop a truck classification model on the basis of FHWA as well as the body configuration scheme. In this dissertation, a new vehicle point cloud reconstruction framework with ground plane consideration was first designed to merge multiple consecutive frames originating from the same vehicle point cloud and to further

obtain a dense representation of each vehicle. Then, the low profile of the vehicle point cloud was extracted and used to classify vehicles based on the FHWA classification scheme. Further, the reconstructed point cloud was adopted as input to a deep representation learning algorithm – PointNet- to classify trucks according to their detailed body configurations. The new LiDAR-based truck classification method was able to retain a wide LiDAR detection zone with a promising correct classification rate.

3. A self-learning framework was designed to alleviate the obsolescence of the vehicle truck classification system through the integration of inductive loop sensors and LiDAR sensors, the latter of which has been proven to have the ability to recognize the truck axle configuration. The self-learning framework adopted the LiDAR-based classification model as a data labeling platform to produce reliable FHWA class labels for validating and upgrading the legacy signature-based classification model, which may not be optimal in classifying newer trucks operating on the highway over time, since the signature-based classification model does not directly capture the axle configuration of trucks, but rather infers their FHWA classes based on inductive signature characteristics. Subsequently, an adaptive transfer learning framework was developed to improve the performance of the new signature-based classification model without retraining the model with the overall dataset and to further reduce the computation burden. In the near future, conventional diesel and gasoline trucks may transition towards zero-emission (ZE) fuel cell or battery-electric trucks which may possess different truck body characteristics. The designed self-learning framework

possesses the potential to accommodate such changes and produce reliable FHWA classification data while the newer eco-friendly ZE trucks penetrate the market.

## **5.2 Future Work**

Trucks can be characterized into different categories based on their diverse physical features (e.g. axle configurations, body configurations, weights, models, etc.) for the use of various transportation applications. Therefore, a single characterization scheme is never sufficient to provide a comprehensive understanding of truck movements and further address trucking-related concerns.

One potential future direction of this dissertation is to leverage the existing sensor setup to provide much more detailed truck class information, such as the gross vehicle weight rating classification which can be directly used for emissions analysis, and in freight forecasting.

In addition, the designed truck monitoring system can be further extended through fusion with tracking sensors, such as Global Positioning System (GPS) and Automated License Plate Readers (ALPR), to provide truck trajectories by their detailed classifications and further derive the truck origin-destination information to better support the task of freight forecasting in the future.

## REFERENCES:

- AASHTO. (1993). *AASHTO Guide for Design of Pavement Structure*.
- Abdelbaki, H. M., Hussain, K., & Gelenbe, E. (2001). A laser intensity image-based automatic vehicle classification system. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 460–465. <https://doi.org/10.1109/itsc.2001.948701>
- Allu, K., Sun, Z., & Tok, A. (2020). LiDAR-based Reconstruction for Truck Surveillance. *UC Irvine: Institute of Transportation Studies*.  
<https://escholarship.org/uc/item/68b3g336>
- Aouatef Mahanil; Ahmed Riad Baba Ali. (2016). Classification Problem in Imbalanced Datasets. In *Intech: Vol. I* (Issue tourism, p. 13).  
<https://doi.org/http://dx.doi.org/10.5772/57353>
- Asborn, M. I., Burris, C. G., & Hernandez, S. (2019). Truck Body-Type Classification using Single-Beam Lidar Sensors. *Transportation Research Record*, 2673(1), 26–40.  
<https://doi.org/10.1177/0361198118821847>
- Aschauer, D. A. (1990). Highway capacity and economic growth. *Economic Perspectives*, 14–24.
- Athol, P. (1965). Interdependence of Certain Operational Characteristics within a Moving Traffic Stream. *Highway Research Record*, 72, 58–87.  
<http://trid.trb.org/view.aspx?id=117031>
- Bachmann, C., Roorda, M. J., Abdulhai, B., & Moshiri, B. (2013). Fusing a Bluetooth traffic monitoring system with loop detector data for improved freeway traffic speed estimation. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 17(2), 152–164. <https://doi.org/10.1080/15472450.2012.696449>
- Beagan, D., Tempesta, D., & Proussaloglou, K. (2019). *Quick Response Freight Methods (QRFM)*. <https://ops.fhwa.dot.gov/publications/fhwahop19057/fhwahop19057.pdf>
- Besl, P., & McKay, N. (1992). A Method for Registration of 3D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 239–256.
- Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). Simple online and real-time tracking. *Proceedings - International Conference on Image Processing, ICIP, 2016-August*, 3464–3468. <https://doi.org/10.1109/ICIP.2016.7533003>
- Bitar, N., & Refai, H. H. (2017). A Probabilistic Approach to Improve the Accuracy of Axle-Based Automatic Vehicle Classifiers. *IEEE Transactions on Intelligent Transportation Systems*, 18(3), 537–544. <https://doi.org/10.1109/TITS.2016.2580058>
- Breiman, L. (1994). Bagging predictors: Technical Report No. 421. *Department of Statistics University of California*, 2, 19.  
<https://www.stat.berkeley.edu/%7B~%7Dbreiman/bagging.pdf>
- Cambridge Systematics Inc., & Battelle Memorial Institute. (2005). *An Initial Assessment of Freight Bottlenecks on Highways*. October, 191.

- Chen, Y., & Medioni, G. (1991). Object Modeling by Registration of Multiple Range Images. In *Proceedings - IEEE International Conference on Robotics and Automation* (pp. 2724–2729). <https://graphics.stanford.edu/~smr/ICP/comparison/chen-medioni-align-rob91.pdf>
- Choi, S., Zhou, Q. Y., & Koltun, V. (2015). Robust reconstruction of indoor scenes. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 07-12-June*, 5556–5565. <https://doi.org/10.1109/CVPR.2015.7299195>
- Coifman, B. (2001). Improved velocity estimation using single loop detectors. *Transportation Research Part A: Policy and Practice*, 35(10), 863–880. [https://doi.org/10.1016/S0965-8564\(00\)00028-8](https://doi.org/10.1016/S0965-8564(00)00028-8)
- Coifman, B., Dhoorjaty, S., & Lee, Z. H. (2003). Estimating median velocity instead of mean velocity at single loop detectors. *Transportation Research Part C: Emerging Technologies*, 11(3–4), 211–222. [https://doi.org/10.1016/S0968-090X\(03\)00025-1](https://doi.org/10.1016/S0968-090X(03)00025-1)
- Coifman, B., & Kim, S. B. (2009). Speed estimation and length-based vehicle classification from freeway single-loop detectors. *Transportation Research Part C: Emerging Technologies*, 17(4), 349–364. <https://doi.org/10.1016/j.trc.2009.01.004>
- Coifman, B., & Neelisetty, S. (2004). Improved single loop velocity estimation in the presence of heavy truck traffic. *Transportation Research Board*. <https://doi.org/10.1080/15472450.2013.801708>
- Coifman, B., & Neelisetty, S. (2014). Improved speed estimation from single-loop detectors with the high truck flow. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 18(2), 138–148. <https://doi.org/10.1080/15472450.2013.801708>
- Daganzo, C. F. (2002). A behavioral theory of multi-lane traffic flow. Part I: Long homogeneous freeway sections. *Transportation Research Part B: Methodological*, 36(2), 131–158. [https://doi.org/10.1016/S0191-2615\(00\)00042-4](https://doi.org/10.1016/S0191-2615(00)00042-4)
- Daszykowski, M., & Walczak, B. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd*, 96, 226–231. <https://doi.org/10.1016/B978-044452701-1.00067-3>
- Dong, C., Richards, S. H., Huang, B., & Jiang, X. (2015). Identifying the factors contributing to the severity of truck-involved crashes. *International Journal of Injury Control and Safety Promotion*, 22(2), 116–126. <https://doi.org/10.1080/17457300.2013.844713>
- Efron, B. (1988). Bootstrap Methods: Another Look at the Jackknife. *Annals of Statistics*, 2(5), 347–370. <http://projecteuclid.org/euclid.aop/1176996548>
- Evangelidis, G. D., Kounades-Bastian, D., Horaud, R., & Psarakis, E. Z. (2014). A generative model for the joint registration of multiple point sets. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8695 LNCS(PART 7), 109–122. [https://doi.org/10.1007/978-3-319-10584-0\\_8](https://doi.org/10.1007/978-3-319-10584-0_8)

- Federal Highway Administration. (2013). *Traffic Monitoring Guide FHWA* (Issue October). <http://www.fhwa.dot.gov/policyinformation/tmguide/>
- Federal Highway Administration. (2019). *Compilation of Existing State Truck Size and Weight Limit Laws*. [https://ops.fhwa.dot.gov/freight/policy/rpt\\_congress/truck\\_sw\\_laws/app\\_a.htm](https://ops.fhwa.dot.gov/freight/policy/rpt_congress/truck_sw_laws/app_a.htm)
- FHWA (Federal Highway Administration). (2014). *Verification, refinement, and applicability of long-term pavement performance vehicle classification rules: chapter 2. introduction to vehicle classification*. <https://www.fhwa.dot.gov/publications/research/infrastructure/pavements/ltp/13091/002.cfm>
- Gao, W., & Tedrake, R. (2019). FilterReg: Robust and efficient probabilistic point-set registration using Gaussian filter and twist parameterization. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2019-June*, 11087–11096. <https://doi.org/10.1109/CVPR.2019.01135>
- Gillespie, T. D., Karamihas, S. M., & Sayer, M. W. (1993). *Effects of Heavy-Vehicle Characteristics on Pavement Response and Performance (NCHRP Report 353)*.
- Gillespie, T., & Karamihas, S. (2009). Heavy Truck Properties Significant to Pavement Damage. *Vehicle-Road Interaction*, 52-52–12. <https://doi.org/10.1520/stp13248s>
- Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Journal of Machine Learning Research*, 9, 249–256.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. [http://www.deeplearningbook.org/front\\_matter.pdf](http://www.deeplearningbook.org/front_matter.pdf)
- Guensler, R., Yoon, S., Li, H., & Jun, J. (2005). *Heavy-Duty Diesel Vehicle Modal Emission Model (HDDV-MEM) Volume I: Modal Emission Modeling Framework: Vol. I* (Issue August).
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *Proceedings of the IEEE International Conference on Computer Vision, 2015 Inter*, 1026–1034. <https://doi.org/10.1109/ICCV.2015.123>
- Hernandez, S. V. (2014). *Integration of Weigh-In-Motion and Inductive Signature Data for Truck Body Classification*. <https://cloudfront.escholarship.org/dist/prd/content/qt39s3k7bw/qt39s3k7bw.pdf>
- Hernandez, S. V. (2016). Integration of Weigh-in-Motion and Inductive Signature Data for Truck Body Classification. *Transportation Research Part C: Emerging Technologies*, 68, 21.
- Hernandez, S. V., Tok, A., & Ritchie, S. G. (2016). Integration of Weigh-in-Motion (WIM) and inductive signature data for truck body classification. *Transportation Research Part C: Emerging Technologies*, 68, 1–21. <https://doi.org/10.1016/j.trc.2016.03.003>
- Hoeting, J. A., Madigan, D., Raftery, A. E., Volinsky, C. T., Raferty, A. E., & Volinshy, C. T.

- (1999). Bayesian Model Averaging: A Tutorial (with discussion). *Statistical Science*, 14(4), 382–417.
- Horraud, R., Forbes, F., Yguel, M., Dewaele, G., & Zhang, J. (2011). Rigid and articulated point registration with expectation conditional maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3), 587–602.  
<https://doi.org/10.1109/TPAMI.2010.94>
- Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5), 359–366.  
[https://doi.org/10.1016/0893-6080\(89\)90020-8](https://doi.org/10.1016/0893-6080(89)90020-8)
- Hussain, K. F., & Moussa, G. S. (2005). Laser Intensity Vehicle Classification System based on random neural network. *Proceedings of the Annual Southeast Conference*, 1, 131–135.  
<https://doi.org/10.1145/1167350.1167372>
- Jeng, S. T. (2007). *Real-time vehicle reidentification system for freeway performance measurements* [University of California, Irvine]. <https://doi.org/10.11436/mssj.15.250>
- Jeng, S. T., Chu, L., & Hernandez, S. (2013). Wavelet-k nearest neighbor vehicle classification approach with inductive loop signatures. *Transportation Research Record*, 2380, 72–80. <https://doi.org/10.3141/2380-08>
- Jeng, S. T., & Ritchie, S. G. (2008). Real-time vehicle classification using inductive loop signature data. *Transportation Research Record*, 2086, 8–22.  
<https://doi.org/10.3141/2086-02>
- Jian, B., Society, I. C., & Vemuri, B. C. (2011). Robust Point Set Registration Using Gaussian Mixture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8), 1633–1645.
- Joshi, M. V. (2002). On evaluating the performance of classifiers for rare classes. *Proceedings - IEEE International Conference on Data Mining, ICDM*, 641–644.  
<https://doi.org/10.1109/icdm.2002.1184018>
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering, Transactions of the ASME*, 82(1), 35–45.  
<https://doi.org/10.1115/1.3662552>
- Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–15.
- Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4), 221–232.  
<https://doi.org/10.1007/s13748-016-0094-0>
- Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics*, 2(1–2), 83–97. <https://doi.org/10.1002/nav.20053>
- Kwigizile, V., Mussa, R. N., & Selekwa, M. (2005). Connectionist Approach to Improving



- Highway Vehicle Classification Schemes - The Florida Case. *Journal of the Transportation Research Board*, 1917, 182–189.
- Lao, Y., Zhang, G., Corey, J., & Wang, Y. (2012). Gaussian Mixture Model-Based Speed Estimation and Vehicle Classification Using Single-Loop Measurements. *Journal of Intelligent Transportation Systems*, 16(4), 184–196. <https://doi.org/10.1080/15472450.2012.706196>
- Lee, H., & Coifman, B. (2012). Side-fire lidar-based vehicle classification. *Transportation Research Record*, 2308, 173–183. <https://doi.org/10.3141/2308-19>
- Li, Y., Allu, K., Sun, Z., Tok, A., Feng, G., & Ritchie, S. (2021). An Ensemble Approach to Truck Body Type Classification using Deep Representation Learning on 3D Point Sets. *Transportation Research Board 100th Annual Meeting*.
- Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *ArXiv, Section 2*, 1–10.
- Ma, W., Xing, D., McKee, A., Bajwa, R., Flores, C., Fuller, B., & Varaiya, P. (2014). A wireless accelerometer-based automatic vehicle classification prototype system. *IEEE Transactions on Intelligent Transportation Systems*, 15(1), 104–111. <https://doi.org/10.1109/TITS.2013.2273488>
- Ma, X., McCormack, E. D., & Wang, Y. (2011). Processing commercial global positioning system data to develop a web-based truck performance measures program. *Transportation Research Record*, 2246, 92–100. <https://doi.org/10.3141/2246-12>
- Malyshkina, N., & Mannering, F. (2008). Effect of Increases in Speed Limits on Severities of Injuries in Accidents. *Transportation Research Record: Journal of the Transportation Research Board*, 2083, 122–127. <https://doi.org/10.3141/2083-14>
- Mehar, A., Chandra, S., & Velmurugan, S. (2013). Speed and acceleration characteristics of different types of vehicles on multi-lane highways. *European Transport - Trasporti Europei*, 55, 1–12.
- Mele, B., & Altarelli, G. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Physics Letters B*, 299(3–4), 345–350. [https://doi.org/10.1016/0370-2693\(93\)90272-J](https://doi.org/10.1016/0370-2693(93)90272-J)
- Modha, D. S., & Spangler, W. S. (2003). Feature Weighting in k-Means Clustering. *Machine Learning*, 52(3), 217–237. <https://doi.org/10.1023/A:1024016609528>
- Myronenko, A., & Song, X. (2010). Point set registration: Coherent point drifts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12), 2262–2275. <https://doi.org/10.1109/TPAMI.2010.46>
- Nair, V., & Hinton, G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. *Proceedings of the 27th International Conference on Machine Learning*, 3, 807–814. <https://doi.org/10.1.1.165.6419>
- Oh, S., Ritchie, S., & Oh, C. (2002). Real-Time Traffic Measurement from Single Loop

- Inductive Signatures. *Transportation Research Record: Journal of the Transportation Research Board*, 1804(02), 98–106. <https://doi.org/10.3141/1804-14>
- Oppenheim, V. A., Schafer, W. R., & Buck, R. J. (1999). *Discrete-Time Signal Processing* (M. Horton (ed.); 2nd ed.). Prentice-Hall, Inc.
- Park, S., & Ritchie, S. G. (2010). Innovative single-loop speed estimation model with advanced loop data. *IET Intelligent Transport Systems*, 4(4), 232. <https://doi.org/10.1049/iet-its.2009.0126>
- Pomerleau, F., Colas, F., & Siegwart, R. (2015). A Review of Point Cloud Registration Algorithms for Mobile Robotics. *A Review of Point Cloud Registration Algorithms for Mobile Robotics*. <https://doi.org/10.1561/9781680830255>
- Pomerleau, F., Colas, F., Siegwart, R., Magnenat, S., Pomerleau, F., Colas, F., Siegwart, R., Magnenat, S., Icp, C., & Roland, F. C. (2013). Comparing ICP Variants on Real-World Data Sets. *Autonomous Robots*, 34(3), 133–148.
- Pushkar, A., & Acha-daza, J. A. (1994). Estimation of Speeds from Single-Loop Freeway Flow and Occupancy Data Using Cusp Catastrophe Theory Model. *0*(2), 149–157.
- Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017). PointNet: Deep learning on point sets for 3D classification and segmentation. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-Janua*, 77–85. <https://doi.org/10.1109/CVPR.2017.16>
- Quinley, R. (2010). *WIM Data Analyst's Manual: FHWA Report IF-10-018*. 183.
- Raftery, A. E., Gneiting, T., Balabdaoui, F., & Polakowski, M. (2005). Using Bayesian model averaging to calibrate forecast ensembles. *Monthly Weather Review*, 133(5), 1155–1174. <https://doi.org/10.1175/MWR2906.1>
- Ruey Long Cheu, Der-Horng Lee, & Chi Xie. (2001). An arterial speed estimation model fusing data from stationary and mobile sensors. *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, 573–578. <https://doi.org/10.1109/ITSC.2001.948723>
- Sahin, O., Nezafat, R. V., & Cetin, M. (2020). Methods for classification of truck trailers using side-fire light detection and ranging (LiDAR) Data. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 0(0), 1–13. <https://doi.org/10.1080/15472450.2020.1733999>
- Sandhawalia, H., Rodriguez-Serrano, J. A., Poirier, H., & Csurka, G. (2013). Vehicle type classification from laser scanner profiles: A benchmark of feature descriptors. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 517–522. <https://doi.org/10.1109/ITSC.2013.6728283>
- Schaefer, Ron; Worth, Monica; Heilman, Jonathan; Kehoe, N. (2017). Freight Demand Modeling and Data Improvement. In *Freight Demand Modeling and Data Improvement* (Issue December). <https://doi.org/10.17226/22734>

- Schmoyer, R., & Hu, P. S. (1998). Analysis of Vehicle Classification and Truck Weight Data of the New England States. *System, September*.
- Sharman, B. W., & Roorda, M. J. (2011). Analysis of freight global positioning system data: Clustering approach for identifying trip destinations. *Transportation Research Record, 2246*, 83–91. <https://doi.org/10.3141/2246-11>
- Shiyu Liang; R. Srikant. (2017). Why Deep Neural Networks for Function Approximation? *ICLR*. <https://doi.org/10.1109/nnspp.1995.514875>
- Statistics Bureau of Transportation. (2021). *National Transportation Statistics*. <https://www.bts.gov/product/national-transportation-statistics>
- Sun, C., & Ritchie, S. G. (1999). Individual Vehicle Speed Estimation Using Single Loop Inductive Waveforms. *Journal of Transportation Engineering, 125*(December), 531–538.
- Thiagarajan, A., Ravindranath, L., Lacurts, K., & Eriksson, J. (2014). *VTrack : Accurate, energy-aware road traffic delay estimation using mobile phones* Citation Accessed Citable Link Detailed Terms VTrack : Accurate, Energy-aware Road Traffic Delay Estimation Using Mobile Phones.
- Tok, A., Hernandez, S., & Ritchie, S. G. (2009). Accurate individual vehicle speeds from single inductive loop signatures. *Transportation Research Board 88th Annual Meeting*, 1–22.
- Tok, A., Hyun, K. (Kate), Hernandez, S., Jeong, K., Sun, Y. (Ethan), Rindt, C., & Ritchie, S. G. (2017). Truck Activity Monitoring System for Freight Transportation Analysis. *Transportation Research Record: Journal of the Transportation Research Board, 2610*, 97–107. <https://doi.org/10.3141/2610-11>
- U.S. Department of Transportation. (2019). *Commercial Motor Vehicle - Traffic Safety Facts. July*. <http://www-nrd.nhtsa.dot.gov/Pubs/811620.pdf><http://www-nrd.nhtsa.dot.gov/Pubs/809778.pdf>
- U.S. Department of Transportation, Bureau of Transportation Statistics, & U.S. Department of Commerce. (2017). *2017 Commodity Flow Survey* (Issue September 2020).
- United States Environmental Protection Agency. (n.d.). *MOVES2014a: Latest Version of MOtor Vehicle Emission Simulator (MOVES)*. Retrieved July 29, 2018, from <https://www.epa.gov/moves/moves2014a-latest-version-motor-vehicle-emission-simulator-moves>
- Vatani Nezafat, R., Sahin, O., & Cetin, M. (2019). Transfer Learning Using Deep Neural Networks for Classification of Truck Body Types Based on Side-Fire Lidar Data. *Journal of Big Data Analytics in Transportation, 1*(1), 71–82. <https://doi.org/10.1007/s42421-019-00005-9>
- Velodyne Acoustics Inc. (2018). *VLP-32C User Manual*.
- Vo, T. (2011). *An Investigation of Bluetooth Technology for Measuring Travel Times on*

*Arterial Roads: A Case Study on Spring Street* (Issue May). Georgia Institute of Technology.

- Wang, Z., Goodchild, A., & McCormack, E. (2016). Measuring Truck Travel Time Reliability Using Truck Probe GPS Data. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 20(2), 103–112. <https://doi.org/10.1080/15472450.2014.1000455>
- Wilcoxon, F. (1946). Individual comparisons of grouped data by ranking methods. *Journal of Economic Entomology*, 39(6), 269. <https://doi.org/10.1093/jee/39.2.269>
- Wu, J., Xu, H., Zheng, Y., Zhang, Y., Lv, B., & Tian, Z. (2019). Automatic Vehicle Classification using Roadside LiDAR Data. *Transportation Research Record*, 2673(6), 153–164. <https://doi.org/10.1177/0361198119843857>
- Yang, R. (2009). *Vehicle Detection and Classification from a LIDAR-equipped probe vehicle*.
- Zanjani, A. B. (2014). *Deriving Statewide Freight Truck Flows from Global Positioning System (GPS) Data*. January.
- Zanjani, A. B., Pinjari, A. R., Kamali, M., Thakur, A., Short, J., Mysore, V., & Tabatabaee, S. F. (2015). Estimation of Statewide Origin–Destination Truck Flows From Large Streams of GPS Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2, 87–96. <https://doi.org/10.3141/2494-10>
- Zhou, Q. Y., Park, J., & Koltun, V. (2018). Open3D: A modern library for 3D data processing. *ArXiv*.
- Zhou, Z. H., Wu, J., & Tang, W. (2002). Ensembling neural networks: Many could be better than all. *Artificial Intelligence*, 137(1–2), 239–263. [https://doi.org/10.1016/S0004-3702\(02\)00190-X](https://doi.org/10.1016/S0004-3702(02)00190-X)